UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

EMULADOR E MONTADOR PARA O MICRO PCIR

por

VERA MARIA DOS SANTOS ALVES

Dissertação submetida como requisito parcial para
a obtenção do grau de Mestre em
Ciência da Computação

Prof. Tadeu Botteri Corso
orientador

Prof. Antônio Carlos da Rocha Costa
co-orientador

Porto Alegre, março de 1987.

CATALOGAÇÃO NA FONTE

Alves, Vera Maria dos Santos
Emulador e montador para o micro PCIR. Porto Alegre, PGCC da UFRGS, 1987.
iv.
Diss. (mestr. ci. comp.) UFRGS-PGCC, Porto alegre, BR-RS, 1987.
Dissertação: Emulador
Montador

AGRADECIMENTOS

Este trabalho foi realizado, graças à colaboração de muitas pessoas, as quais classifico como fazendo parte da "equipe técnica" que contribuiu através de seus conhecimentos, ou da "equipe de apoio" que sem conhecimentos técnicos, proporcionou o estímulo imprescindível ao longo da tarefa.

Fazendo parte da "equipe técnica", devo agradecer especialmente:

Ao professor Antonio Carlos da Rocha Costa, pela motivação ao estudo da microprogramação que deu origem ao presente trabalho.

Ao professor Thadeu Botteri Corso, pela orientação e acompanhamento.

Aos demais professores, pelos ensinamentos recebidos e particularmente ao professor Simão Sirineo Toscani pela motivação ao estudo de sistemas operacionais.

Ao colega Todesco por ter oportunizado e incentivado o trabalho.

Ao colega Mario Bastos pela paciência e disponibilidade na assessoria ao equipamento utilizado.

Fazendo parte da "equipe de apoio" gostaria de agradecer em especial:

Ao meu filho Daniel a quem não foi dada a oportunidade de escolha, mas que foi envolvido neste

trabalho, adaptando-se aos cuidados de outras pessoas.

Ao meu esposo Círio, pelo despreendimento e compreensão.

Aos meus pais João e Julieta e meu irmão Rubens que me propiciaram condições favoráveis de estudo, acolhendo carinhosamente meu filho, pela ajuda e incentivo.

À amiga Rose, pela disponibilidade e colaboração.

Fazendo parte das duas equipes, agradeço particularmente à colega e amiga Elenara que tornou cada momento menos difícil, me passando a sua garra e seu otimismo.

Finalmente quero externar meus agradecimentos ao Curso de Pós-Graduação em Ciência da Computação, pelas condições oferecidas e à Universidade do Vale do Rio dos Sinos, pela disponibilidade de instalações e equipamentos.

À Daniel e
Cirio,

com amor.

SUMÁRIO

LISTA DE ABREVIATURAS

barram.	barramento
CF	código fonte
CI	código intermediário
CO	código objeto
dest.	destino
ender.	endereço
Obs.	Observação
orig.	origem
RAM	RANDOM-ACESS MEMORY
reg.	registrador
ROM	READ-ONLY MEMORY
ROT.	RÓTULO
TD	tabela de diretivas
TM	tabela de instruções de máquina
TR	tabela de referências cruzadas
TS	tabela de símbolos

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Este trabalho descreve o projeto e implementação de um emulador desenvolvido em firmware no minicomputador ED 311 para o microprocessador de 16 bits, denominado PCIR (Processador de Conjunto de Instruções Reduzido) projetado e construido no Curso de Pós-Graduação em Ciência da Computação da UFRGS.

O presente volume contém ainda a especificação e os procedimentos de implementação de um montador de programas PCIR escrito na própria linguagem de montagem PCIR, aqui definida.

Estes componentes foram desenvolvidos paralelamente ao desenvolvimento do hardware dando subsídios para a avaliação da própria máquina em projeto.

ABSTRACT

This dissertation describes the design and implementation of a firmware emulator developed in the ED 311 minicomputer, for a 16 bits microprocessor, called PCIR (Processador de Conjunto de Instruções Reduzido) designed and constructed at UFRGS (Universidade Federal do Rio Grande do Sul).

The present text also presents the especification and the implementation of a PCIR assembler, written in the own PCIR assembler language.

These software/firmware components were developed simultaneouly to the hardware, in order to aid for in the evaluation of the machine design.

1 INTRODUÇÃO

Este trabalho originou-se a partir do projeto de um microprocessador de 16 bits do tipo RISC (Reduced Instruction Set Computer) /TOD 86/, denominado PCIR (Processador de Conjunto de Instruções Reduzido) que está sendo desenvolvido no Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Rio Grande do Sul.

É comum ser construído junto com o próprio microprocessador, um sistema de desenvolvimento cuja função é assistir o projetista na elaboração do software.

O objetivo primeiro deste trabalho é viabilizar a construção paralela de software e hardware, tornando a máquina PCIR disponível antes da conclusão de seu projeto físico.

A fim de simular o meio ambiente final, o PCIR foi emulado no minicomputador ED 311 e é acessado através de um gerente de módulos objeto que é responsável pela carga dos programas PCIR e pela ativação do emulador. Este módulo foi incorporado ao emulador como uma espécie de camada externa à máquina para interagir com o usuário.

O emulador possibilita não só antecipar o desenvolvimento de software como também avaliar o comportamento da própria máquina em projeto.

Durante a execução das instruções, o emulador é passível de ser monitorado para contabilizar o número de vezes que cada tipo de instrução é executada. Isso é feito através de um monitor embutido no emulador e que pode ser

acionado juntamente com a máquina emulada.

O esquema de entrada e saída foi mapeado na memória principal do ED 311. E os procedimentos foram escritos de forma a invocar as rotinas de comunicação com os periféricos as quais fazem parte do depurador de microprogramas que dá suporte ao desenvolvimento de firmware.

Estando a máquina PCIR implementada por emulação, a próxima etapa é facilitar a tarefa de desenvolvimento de software, dotando a máquina de pelo menos um tradutor (montador ou compilador).

Em face a estas duas alternativas de tradutores, levando-se em conta a complexidade e extensão dos compiladores para serem desenvolvidos nesta fase do projeto, optou-se pela construção de um montador escrito na própria linguagem de montagem do PCIR.

A partir das instruções de máquina foi definido um conjunto de instruções simbólicas onde os mnemônicos seguem a proposição de "Uma Linguagem de Montagem Padrão para os Microprocessadores", definida em /FIS 79/.

O montador PCIR consiste de um programa tradutor de duas passagens que aceita como entrada um programa na linguagem simbólica mencionada acima e gera um programa em linguagem de máquina PCIR, executável pelo emulador.

O software aqui desenvolvido oferece um ambiente mínimo para que a máquina PCIR possa ser usada de forma razoável visando a continuidade da implementação.

O capítulo 2 aborda os aspectos básicos da arquitetura e do conjunto de instruções do microprocessador PCIR.

O capítulo 3 apresenta o projeto do emulador, descrevendo inicialmente as características e recursos do ED 311 utilizados no mapeamento da máquina alvo na máquina hospedeira. Encontram-se ainda neste capítulo os procedimentos de emulação da máquina PCIR com ênfase nas rotinas das instruções, implementadas em firmware.

O capítulo 4 descreve o processo de monitoração da execução das instruções do PCIR, objetivando a contabilização das instruções para permitir a sua avaliação.

No capítulo 5 apresenta-se as características funcionais e os procedimentos de implementação do gerente de módulos objeto para a máquina PCIR, através do qual é realizada a comunicação com o usuário e é definido o ambiente PCIR enquanto emulado.

O capítulo 6 define a estrutura do montador PCIR, apresentando o conjunto de instruções simbólicas e a organização das tabelas utilizadas no processo de montagem.

O capítulo 7 é dedicado a descrever e detalhar a implementação do montador PCIR.

O capítulo 8 é o manual do usuário, onde encontram-se os passos necessários para rodar um programa PCIR.

No capítulo 9 são traçadas observações e

conclusões a respeito do software desenvolvido e delineadas algumas sugestões para a continuidade do projeto.

O anexo 1 contém um resumo da linguagem de microprogramação utilizada para escrever o emulador.

O anexo 2 refere-se aos comandos disponíveis no MDE (Micro Depurador Editor) que foram utilizados nos testes do emulador.

O anexo 3 trata da forma de apresentação dos algoritmos no decorrer do trabalho.

O anexo 4 contém a listagem do programa emulador.

2 MICROPROCESSADOR PCIR

Este capítulo descreve as características principais do microprocessador PCIR. Maiores detalhes podem ser encontrados em /TOD 86/.

2.1 Definição

É um microprocessador integrado, do tipo RISC (Reduced Instruction Set Computer), denominado PCIR (Processador com Conjunto de Instruções Reduzido), com palavra de 16 bits, orientado a registradores.

2.2 Arquitetura

As figuras 2.1 e 2.2 apresentam a arquitetura do microprocessador PCIR.

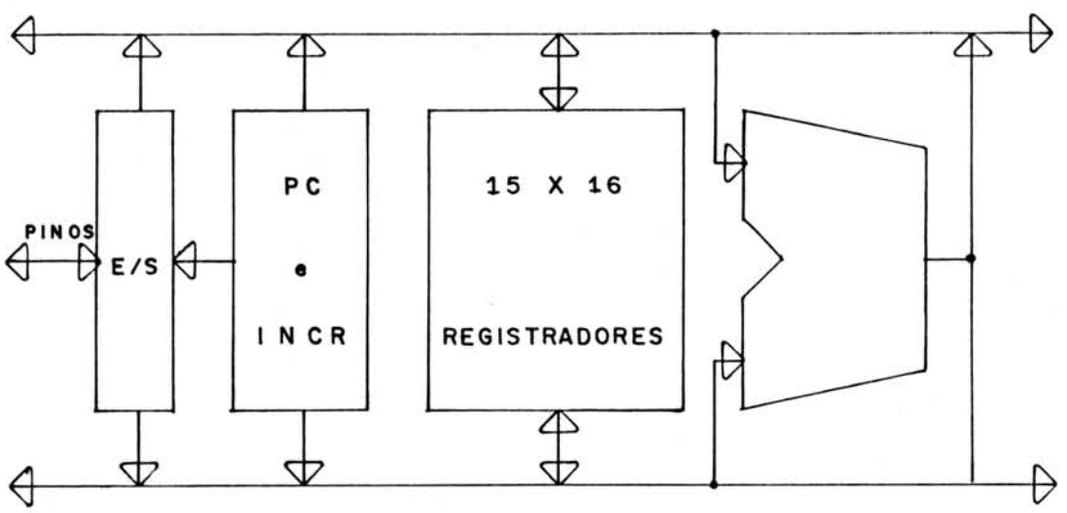
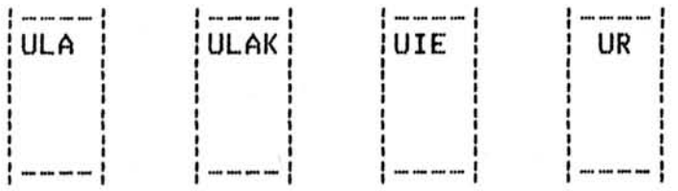


FIGURA 2.1 Diagrama Lógico da Arquitetura do PCIR



ULA - Unidade Aritmética e Lógica

ULAK - Unidade Aritmética e Lógica de Constantes

UIE - Unidade de Inserção e Extração de Bytes

UR - Unidade de rotação

FIGURA 2.2 Parte Operativa do PCIR

2.3 Memória

O microprocessador PCIR tem capacidade para endereçar uma memória de 64 KW (palavra de 16 bits), conforme figura 2.3 onde os endereços são representados em hexadecimal.

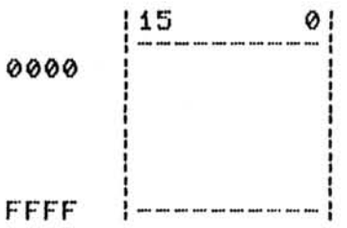


FIGURA 2.3 Memória do PCIR

2.4 Registradores

O microprocessador PCIR apresenta um conjunto de 16 registradores de 16 bits. Sendo R0 a R12, registradores de uso geral e R13 a R15, registradores de uso especial:

- R15 (PC): ponteiro de instruções

- R14 (SP): ponteiro da pilha

- R13 (ST): estados do processador

Os bits indicadores do registrador de estados do processador são afetados internamente pelas operações da unidade aritmética e lógica, unidade aritmética e lógica de constantes e unidade de rotação. E externamente pela requisição de interrupção.

Cada bit indicador tem um significado diferente, conforme especificado na tabela 2.1.

TABELA 2.1 Bits do Registrador de Estados do Processador

BIT	MNEMÔNICO	SIGNIFICADO	ATUALIZAÇÃO
15	N	indica resultado negativo	ULA, ULAK
14	O	indica overflow	ULA,ULAK
13	C	indica "vai-um"	ULA,ULAK,UR
12	I	indica interrupção	SINAL EXTERNO
11	EQ	indica resultado igual a zero	ULA,ULAK
10	GT	indica resultado maior que zero	ULA,ULAK
9	GE	indica resultado maior ou igual a zero	ULA,ULAK
8	LB	indica local de byte	ULA,ULAK (*)

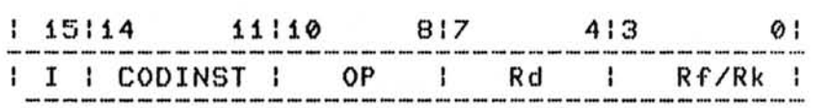
(*) apenas a operação ADDI altera este bit, indicando a posição do byte a ser operado pela UIE.

2.5 Conjunto de Instruções

O microprocessador PCIR , conforme o próprio nome indica apresenta um conjunto de instruções reduzido se comparado com outros microprocessadores. São 12 tipos de instruções de máquina, em dois formatos distintos.

2.5.1 Formato Básico das Instruções

O formato básico das instruções é de uma palavra de 16 bits, onde bit 15 deve ser igual a zero, conforme figura 2.4.



I : indica o formato da instrução
CODINST : código da instrução de máquina
OP : operação da ULA, ULAK, UIE ou UR
Rd : registrador destino
Rf/Rk : registrador fonte ou constante 0, 1, -1

FIGURA 2.4 Formato Básico das Instruções

2.5.2 Formato das Instruções com Valor Imediato

O formato das instruções com valor imediato é de duas palavras, conforme figura 2.5. Os campos da primeira palavra têm significado idêntico aos campos das instruções no formato básico porém o bit 15 deve ser igual a 1.

Im - valor imediato de 16 bits

FIGURA 2.5 Formato das Instruções com Valor Imediato

2.5.3 Instruções de Máquina

As instruções de máquina do PCIR, são mostradas na tabela 2.2 com seu significado nos dois formatos.

TABELA 2.2 Instruções de Máquina

CÓDIGO decimal	CODINST binário	MNEMÔNICO	SIGNIFICADO	
			I=0	I=1
0	0000	OP	Rd<- Rd op Rf	Rd<- Im op Rf
1	0001	OPK	Rd<- Rd op Rk	Rd<- Im op Rk
2	0010	IE	Rd<- Rd ie Rf	Rd<- Im ie Rf
3	0011	SEQ	se Rf=Rd então Rd<- -1 senão Rd<- 0	se Rf=Im então Rd<- -1 senão Rd<- 0
4	0100	SGT	se Rf>Rd então Rd<- -1 senão Rd<- 0	se Rf>Im então Rd<- -1 senão Rd<- 0
5	0101	SGE	se Rf>=Rd então Rd<- -1 senão Rd<- 0	se Rf>=Im então Rd<- -1 senão Rd<- 0
6	0110	BF		se Rf=0 então Rd<- Im
7	0111	RC	Rd<- UR(Rf)	Rd<- UR(Im)
8	1000	STORE	[Rd]<- Rf	[Im]<- Rf
10	1010	PUSH	[Rd]<- Rf Rd<- Rd op 1	[Rd]<- Rf Rd<- Rd op 1 Rf<- Im
12	1100	LOAD	Rd<- [Rs]	Rd<- [Im]
14	1110	POP	Rf<- Rf op 1 Rd<- [Rf]	Rf<- Rf op Im Rd<- [Rf]

Rf - registrador fonte

Rd - registrador destino

Im - valor imediato de 16 bits

Rk - constante especial (0, 1 ou -1)

UR - unidade de rotação

2.5.3.1 Operações da Unidade Aritmética e Lógica

As operações realizadas pela ULA são apresentadas na tabela 2.3.

TABELA 2.3 Operações da ULA

CÓDIGO	OP	MNEMÔNICO	SIGNIFICADO	ST
0	000	ADD	soma	X
1	001	ADDC	soma com "vai-um"	X
2	010	SUB	subtração	X
3	011	SUBC	subtração com "vai-um"	X
4	100	ADDI	soma de índice	X
5	101	AND	E lógico	X
6	110	OR	OU lógico	X
7	111	XOR	OU exclusivo lógico	X

2.5.3.2 Operações da Unidade de Rotação

As operações realizadas pela UR são apresentadas na tabela 2.4.

TABELA 2.4 Operações da UR

CÓDIGO	OP	MNEMÔNICO	SIGNIFICADO	ST
0	000	ROLC	rotação à esquerda com "vai-um"	X
2	010	RORC	rotação à direita com "vai-um"	X
4	100	MOVE	transferência sem rotação	

2.5.3.3 Operações da Unidade de Inserção e Extração de Byte

As operações realizadas pela UIE são apresentadas na tabela 2.5.

TABELA 2.5 Operações da UIE

CÓDIGO	OP	MNEMÔNICO	SIGNIFICADO	ST
0	000	INSB	insere byte	
1	001	EXTB	extrai byte	

3 EMULADOR

Este capítulo apresenta as características da máquina hospedeira, da máquina PCIR emulada e os recursos utilizados na realização do emulador.

3.1 Definição

Segundo Rosin /ROS 69/, emulador é um conjunto de microprogramas residentes em uma memória de controle, que definem uma máquina. A máquina implementada por emulação é conhecida por máquina virtual ou máquina alvo e o computador que executa os microprogramas é denominado máquina hospedeira.

Neste trabalho, a máquina a ser emulada é o microprocessador PCIR (máquina alvo). Para emular o microprocessador está sendo usado como máquina hospedeira, o minicomputador ED 311 por ser microprogramável e possuir suporte razoável para o desenvolvimento de firmware.

3.2 Descrição da Máquina Hospedeira

Para se realizar um emulador deve-se fazer um estudo dos recursos que serão utilizados na emulação, ou seja, aqueles oferecidos pela máquina hospedeira. Isto exige o conhecimento de vários itens, tais como número de registradores, memória disponível e outros. Aqui descrevemos em linhas gerais os recursos utilizados do minicomputador ED 311. Maiores detalhes podem ser encontrados em /PAD 80/.

3.2.1 Memória

- Memória principal (RAM), com capacidade de 64 KB que pode ser endereçada em unidades de dois bytes de acordo com o bit 14 do registrador S. O endereço a acessar deve estar no registrador A.

- Memória de controle com capacidade de 24 KB, sendo 2 KB iniciais em ROM e os demais em RAM. Endereçada em unidades de dois bytes através do registrador A.

- Memória local, consistindo de 16 palavras de 16 bits (LM0 a LM15). O endereço a acessar deve estar no registrador EH.

3.2.2 Registradores e Funções

O processador possui 22 registradores de 16 bits. A configuração dos registradores do ED 311 é ilustrada na figura 3.1.

FIGURA 3.1 Configuração dos Registradores do ED 311

A seguir apresenta-se uma breve descrição de cada um dos registradores:

- **Registradores gerais (GR):**

São 8 registradores de 16 bits identificados com os nomes GR0 a GR7. Estes registradores podem ser usados

como registradores de trabalho.

- Registrador pilha de controle (STK):

São 8 registradores de 16 bits associados a um ponteiro de controle. Este ponteiro organiza os registradores como uma lista circular. Quando é armazenado um dado em STK, o ponteiro é adicionado de 1 e quando é lido um dado, ele é decrementado de 1. Quando o ponteiro vale 7 e é adicionado 1, passa a valer 0.

- Registrador de endereço (A):

Registrador de 16 bits utilizado para endereçar tanto a memória principal como a memória de controle.

- Registrador de expansão e endereçador da memória local (E):

Registrador dividido em duas partes de 8 bits. Os 8 bits mais à esquerda (de 0 a 7) são chamados de EH. Seus bits de 0 a 3 são ignorados e os bits de 4 a 7 são utilizados para endereçar a memória local. Os bits mais à direita (de 8 a 15) são chamados de EL e servem como extensão do registrador A para endereçar a memória principal, quando for especificado endereço virtual.

- Registrador temporário (B):

Registrador de 16 bits usado como registrador temporário pela ULA ou como registrador de dados para o barramento S ou, ainda, como registrador de armazenamento de dados vindos da interface de entrada e saída. O conteúdo do registrador B é exibido permanentemente nas lâmpadas

DISPLAY do painel de operação.

- Registrador de dados (D):

Registrador de 16 bits que funciona junto com o registrador B como registrador de dados da ULA. Funciona ainda como registrador para troca de informações com outros componentes do sistema como memória principal, memória de controle, barramentos J-K e S-D.

- Registrador de estados (S):

Registrador de 16 bits que indica o estado do processador. A descrição de cada estado encontra-se detalhada em /PAD 80/.

- Registrador de estados de portas lógicas (H):

Registrador de 16 bits que indica o estado de alguns sinais recebidos pelo processador, bem como o estado de alguns componentes do sistema.

- Contador de microprograma (MPC):

Registrador de 16 bits que contém o endereço da próxima microinstrução a ser executada.

- Registrador de microinstruções (MIR):

Registrador de 16 bits que contém a microinstrução que está sendo executada no momento.

- Registrador de tempo (T) e contador (C):

Registrador de 16 bits dividido em duas partes de 8 bits. A parte da esquerda (0 a 7) é usada como registrador de tempo. A parte direita (8 a 15) é usada como um registrador contador.

- Registrador externo (EXT) e de máscara (MSK):

Registrador de 16 bits dividido em duas partes de 8 bits. A parte esquerda (0 a 7) é usada como registrador externo. A parte direita (8 a 15) é usada como registrador de mascaramento de microinterrupções.

- Registrador indicador de interrupções (INT):

Registrador de 16 bits que funciona como um indicador da causa de uma microinterrupção.

- Registrador produtor de endereços (TLB):

É uma memória associativa de duas partes (parte A e parte B). Cada parte é constituída de 4 registradores de 16 bits. Estes registradores são numerados de 0 a 3.

Cada registrador da parte A se relaciona com o registrador correspondente da parte B.

Os registradores da parte A contêm endereços virtuais (segmento e página). Os registradores da parte B contêm o endereço da página real e os bits de página inválida e alterada, correspondente ao registrador da parte A. A TLB contém a relação das 4 últimas páginas de memória referidas no sistema.

- Registrador das lâmpadas de controle (CTL):

Registrador de 16 bits associado com as lâmpadas de controle no painel de operação.

- Registrador de controle de chaves (SWC):

Registrador de 16 bits associado com as chaves do painel de operação (bits 8 a 15). Os bits de 0 a 7 contêm o valor correspondente ao posicionamento dos botões "MODE SELECTION" e "PROGRAM REGISTER".

- Registrador das chaves de entrada (SWE):

Registrador de 16 bits relacionados com as 16 chaves de entrada (16 a 31) e lâmpadas (0 a 15), no painel de operação.

- Registrador da lâmpada de mensagem (MSL):

Registrador de 16 bits, sendo cada bit associado com um filamento na formação das duas letras da lâmpada de mensagem.

- Registrador das chaves de função (FLK):

Registrador de 16 bits, relacionado com as lâmpadas e chaves de função do painel de operação.

3.2.3 Barramentos

A ligação entre o processador microprogramável de 16 bits e os 22 registradores é feita através de 4

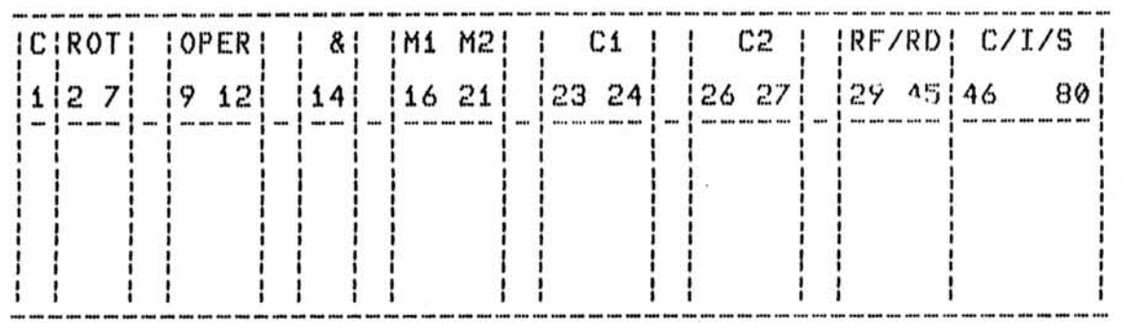
barramentos de 16 bits: D, J, K e S.

3.2.4 Suporte para Firmware

A máquina hospedeira (ED 311) oferece como suporte para o desenvolvimento de firmware, um montador de microprogramas denominado MICMON /EDI 81/ e um depurador de microprogramas, denominado MDE /COR 82/.

3.2.4.1 Micmon

Está disponível no ED 311, o software denominado MICMON o qual pode ser utilizado para montar programas escritos na linguagem de micromontagem descrita no anexo 1. O montador traduz o programa fonte de formato fixo, conforme figura 3.2, gerando o código objeto em disquete.



coluna 1: * indica linha de comentário
Ø indica microinstrução

coluna 2 a 7: rótulo da microiinstrução

coluna 9 a 12: código da operação (ver anexo 1)

coluna 14: R indica leitura na memória
W indica escrita na memória
& indica espera pelo acesso à memória

coluna 16 a 21: modificações (ver anexo 1)

coluna 23 a 24: condição 1

coluna 26 a 27: condição 2

coluna 29 a 45: registradores fonte e destino

coluna 46 a 80: comentário/identificação/seqüência
46 a 72: comentários
73 a 75: identificação
76 a 80: número de seqüência

FIGURA 3.2 Campos do Programa Fonte na Linguagem de Micromontagem

3.2.4.2 MDE

O MDE é um sistema de suporte em firmware para edição e depuração de microprogramas.

O MDE ao ser executado ocupa as posições de mais

alta ordem da memória de controle e não possui mecanismos de proteção contra acessos indevidos do microprograma sob teste. Isto possibilita ao usuário acessar as rotinas do microdepurador.

Os comandos do MDE estão descritos no anexo 2 e suas funções são descritas abaixo:

a) Carga de microprogramas e dados na memória de controle a partir de disco flexível ou disco rígido.

b) Descarga de microprogramas e conteúdo da memória principal em disco flexível ou disco rígido.

c) Impressão de microprogramas ou outras informações a partir da memória principal ou memória de controle.

d) Transferência entre as memórias principal e de controle.

e) Verificação e alteração dos conteúdos da memória principal, memória de controle, registradores, pilha, memória local e tabela de endereços através do terminal.

f) Inserção e supressão de pontos de parada no microprograma sob teste.

g) Execução do microprograma sob teste.

3.3 Mapeamento

O mapeamento foi realizado levando-se em conta o uso do suporte para o desenvolvimento de firmware disponível no ED 311 e a especialização da maioria dos registradores, o que restringe a liberdade do programador de firmware em termos de alocação.

3.3.1 Espaço de endereçamento

O espaço de endereçamento disponível nesta fase fica reduzido em virtude da disponibilidade de memória do ED 311, ou seja, dos 64 KW que a máquina PCIR endereça apenas 32 KW existem na máquina hospedeira, sendo que os endereços iniciais (2 KB) não podem ser usados pelo firmware.

As seções 3.3.2, 3.3.3 e 4.1 especificam a área de memória reservada pelo emulador, ficando então o usuário da máquina PCIR enquanto emulada com memória disponível de aproximadamente 30 KW. O deslocamento inicial ficará transparente ao usuário através de uma rotina do emulador que mapeia endereço virtual (PCIR) em endereço real (ED 311).

A figura 3.3 apresenta o mapeamento da memória PCIR na máquina hospedeira, onde os endereços estão representados em hexadecimal.

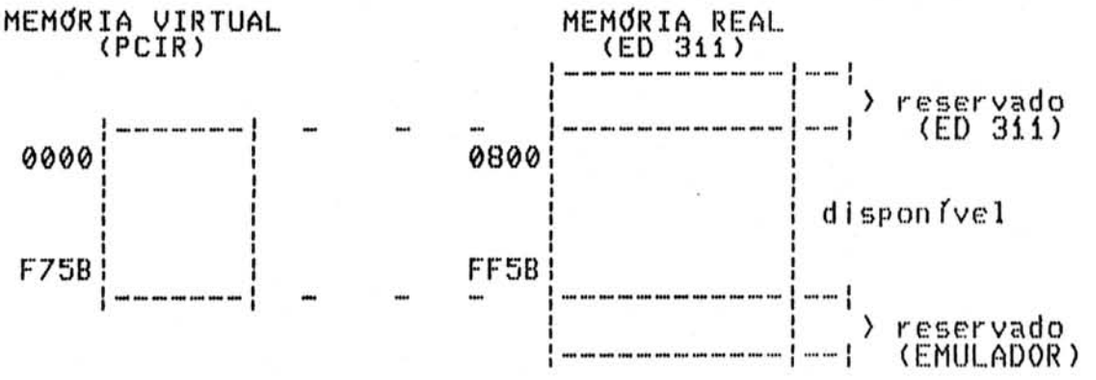


FIGURA 3.3 Espaço de Endereçamento do PCIR emulado

3.3.2 Alocação de Registradores

A alocação dos registradores visou manter as características do microprocessador PCIR o qual não diferencia a referência a qualquer registrador em suas instruções.

O ED 311 não possui registradores de uso geral suficientes para o mapeamento desejado e a memória local que seria adequada a esta alocação tem algumas de suas palavras reservadas pelo MDE.

Para uniformizar o acesso a todos os registradores foram reservadas 16 palavras da memória principal para alocar os registradores do PCIR, conforme figura 3.4.

ENDEREÇO (hexadecimal)	REGISTRADOR
FFE0	R0
FFE2	R1
FFE4	R2
FFE6	R3
FFE8	R4
FFEA	R5
FFEC	R6
FFEE	R7
FFF0	R8
FFF2	R9
FFF4	R10
FFF6	R11
FFF8	R12
FFFA	R13-ST
FFFC	R14-SP
FFFE	R15-PC

FIGURA 3.4 Alocação dos Registradores do PCIR

3.3.3 Entrada e Saída

As rotinas de entrada e saída do MDE sofreram algumas adaptações e foram incorporadas no emulador.

A figura 3.5 mostra a configuração da memória real reservada para entrada e saída, onde os endereços estão representados em hexadecimal.

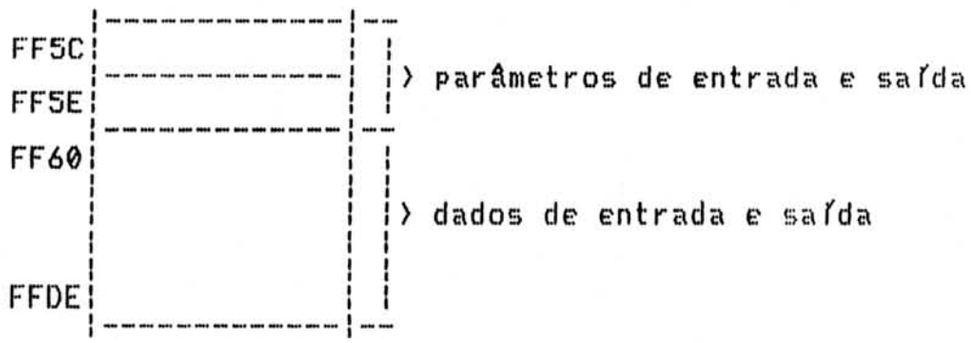


FIGURA 3.5 Entrada e Saída

3.3.4 Operações de Entrada e Saída

As operações de entrada ou saída, bem como os parâmetros necessários a sua execução estarão disponíveis na palavra da memória real FF5C, conforme figura 3.6. A palavra FF5E ficou reservada para futuras expansões.

A cada ciclo o emulador verifica o bit de execução (FF5C<0>) e efetua a operação de entrada ou saída especificada (FF5C<15:13>), enviando os parâmetros para a rotina de entrada ou saída requerida. As operações de entrada e saída disponíveis no emulador são:

a) Leitura em disquete (000)

b) Gravação em disquete (001)

c) Impressão (010)

d) Leitura de teclado (011)

e) Exibição no vídeo (100)

FIGURA 3.6 Configuração da Palavra FF5C

3.3.4.1 Leitura em Disquete

A leitura em disquete é feita por setor. Os parâmetros necessários são a trilha e o setor. Os dados lidos do disquete estarão disponíveis na área da memória principal reservada para os dados de entrada e saída. A figura 3.7 mostra a configuração da palavra de parâmetros quando da leitura em disquete.

FIGURA 3.7 Parâmetros para Leitura em Disquete

3.3.4.2 Gravação em Disquete

A gravação em disquete é feita por setor. Os parâmetros necessários são a trilha de gravação e o setor de gravação. Os dados a serem gravados serão acessados da área da memória principal reservada para os dados de entrada e saída. A figura 3.8 mostra a configuração da palavra de parâmetros quando da gravação em disquete.

0	1 5	6 12	13 15
1	SETOR	TRILHA	001

FIGURA 3.8 Parâmetros para Gravação em Disquete

3.3.4.3 Impressão

A impressão é feita por linha de 128 caracteres, onde o último caractere deve obrigatoriamente ser NL.

A linha a ser impressa é acessada na área da memória principal reservada para os dados de entrada e saída. Os parâmetros necessários à impressão são o controle do carro da impressora, ou seja o número de linhas que devem ser deixadas em branco (SPACE AFTER) e a opção de salto de página (SKIP AFTER) que deve ser 1 se desejada. A figura 3.9 mostra a configuração da palavra de parâmetros quando da impressão de uma linha.

0	1 7	8	9 12	13 15
1	SPACE AFTER	SKIP AFTER		010

FIGURA 3.9 Parâmetros para Impressão

3.3.4.4 Leitura de Teclado

A leitura de teclado é realizada por linha de 40 caracteres. A linha lida fica disponível na área da memória principal reservada para os dados de entrada e saída. Esta operação não exige parâmetros. A figura 3.10 mostra a configuração da palavra de parâmetros, quando da leitura de

teclado.

FIGURA 3.10 Configuração da Palavra de Parâmetros para Leitura de Teclado

3.3.4.5 Exibição no Vídeo

É exibida no vídeo uma linha de até 40 caracteres acessada na área da memória principal reservada para os dados de entrada e saída. Esta operação não exige parâmetros. A figura 3.11 mostra a configuração da palavra de parâmetros quando da exibição no vídeo.

FIGURA 3.11 Configuração da Palavra de Parâmetros quando da Exibição no vídeo

3.4 Emulação do Microprocessador PCIR

O microprocessador PCIR foi emulado por um microprograma cuja listagem encontra-se no anexo 3. Os testes foram efetuados dividindo as instruções em grupos e testando cada grupo separadamente, utilizando-se dos grupos já testados. Estes testes foram realizados via MDE e os comandos para tal estão descritos no anexo 2.

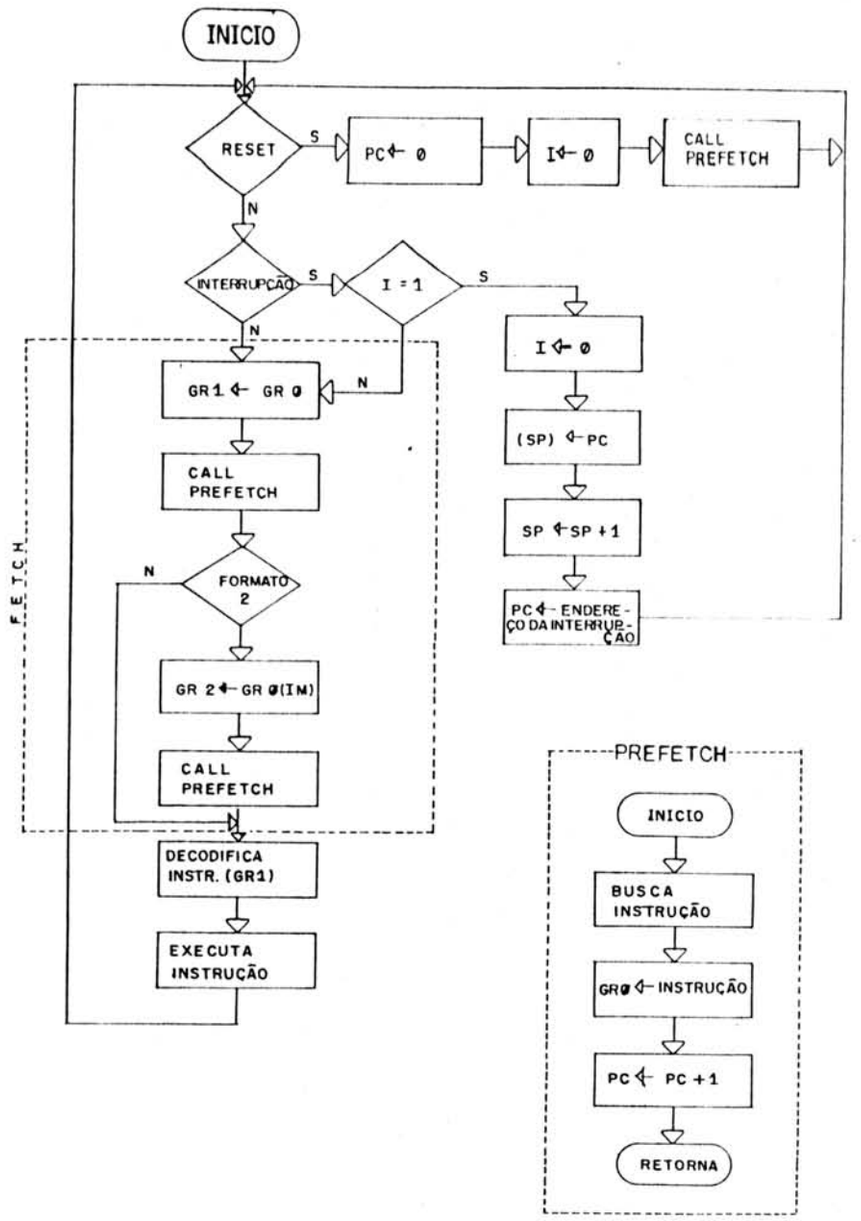
3.4.1 Recursos Utilizados

Para realizar o emulador foi utilizada a linguagem de microprogramação com o respectivo montador descrito na seção 3.2.4.1.

Cada instrução do PCIR será executada por uma rotina em firmware usando os recursos disponíveis na máquina hospedeira.

As próximas seções apresentam o fluxo geral do emulador e as rotinas utilizadas na emulação das instruções do microprocessador PCIR.

3.4.2 Fluxo Geral do Emulador



NOTAÇÃO:

GR0 - Registrador que contém instrução antecipada

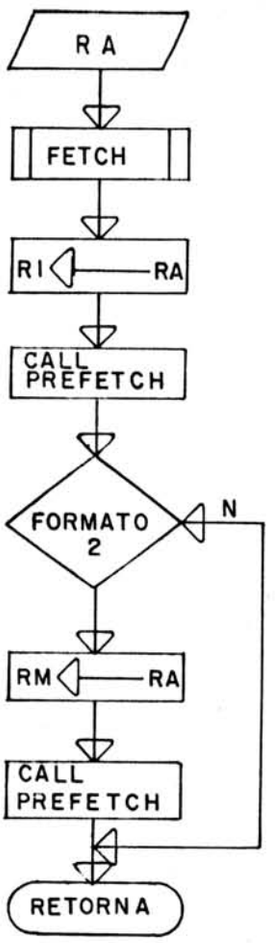
GR1 - Registrador que contém instrução a executar

GR2 - Registrador que contém o operando imediato das instruções de duas palavras

3.4.3 Rotinas Auxiliares

Estas rotinas são chamadas pelas rotinas das instruções. Para facilitar a consulta, são apresentadas no fluxo com o nome usado no programa, utilizando os símbolos:

3.4.3.1 Busca da Instrução



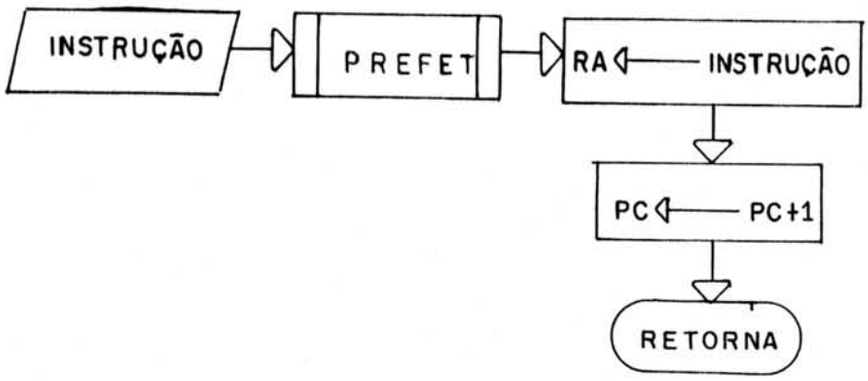
NOTACÃO:

RA - Registrador que contém instrução antecipada

RI - Registrador que contém instrução à executar

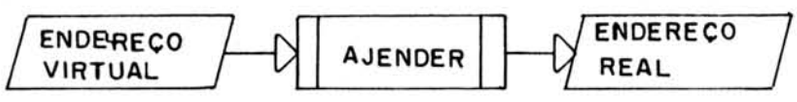
RM - Registrador que contém o operando imediato das instruções de duas palavras

3.4.3.2 Busca Antecipada da Instrução



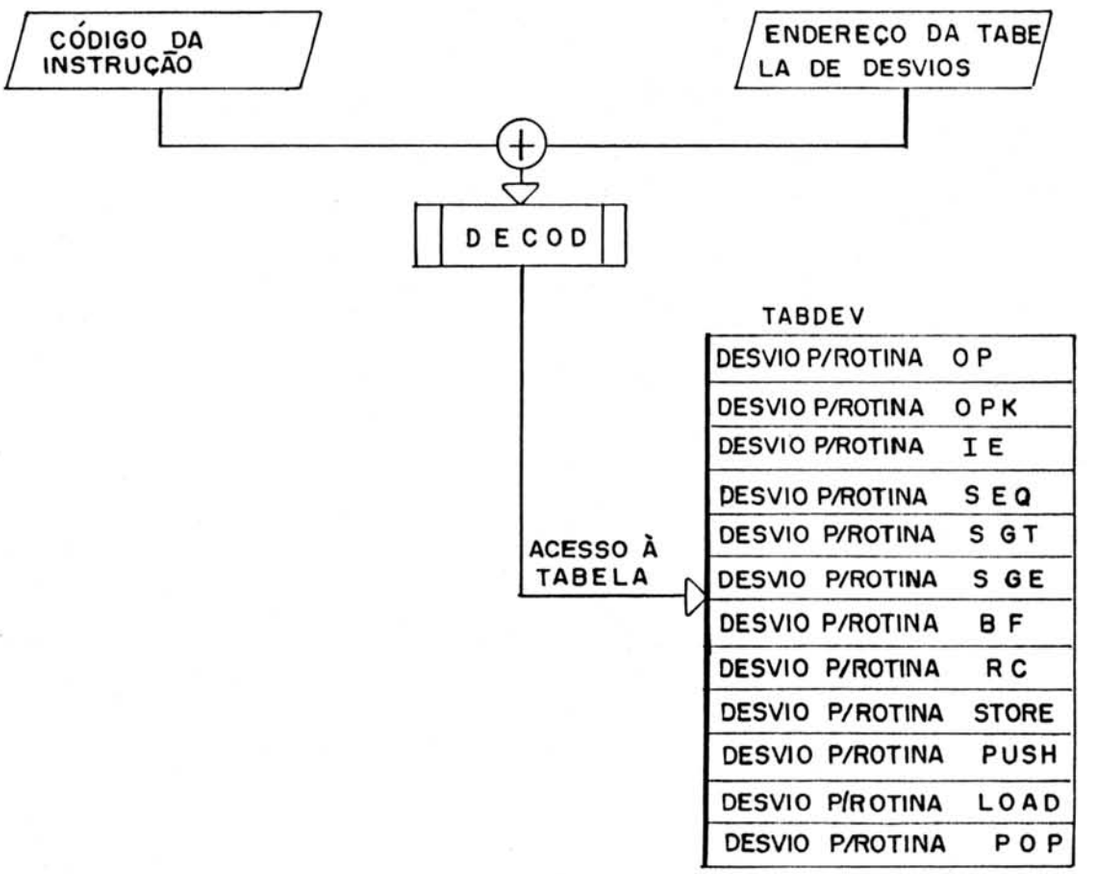
NOTAÇÃO:
RA - Registrador que contém instrução antecipada

3.4.3.3 Ajuste de Endereço

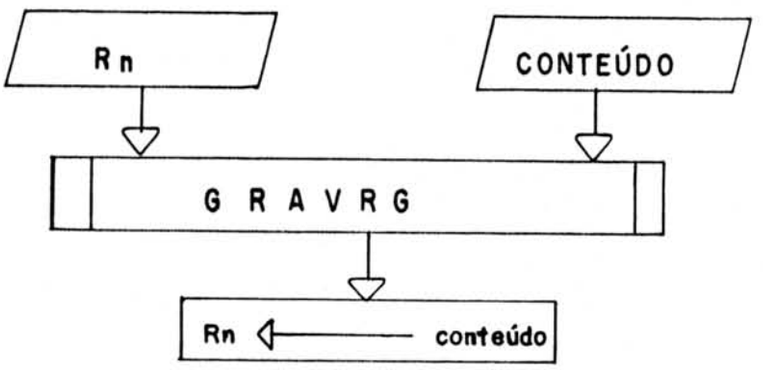


OBS: O endereço é reajustado para acessar endereços pares da memória principal do ED 311 e é deslocado para desviar os 2 KB iniciais.

3.4.3.4 Decodificação das Instruções



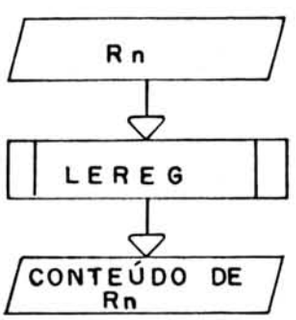
3.4.3.5 Carga em Registrador



NOTAÇÃO:

Rn - Registrador a ser carregado

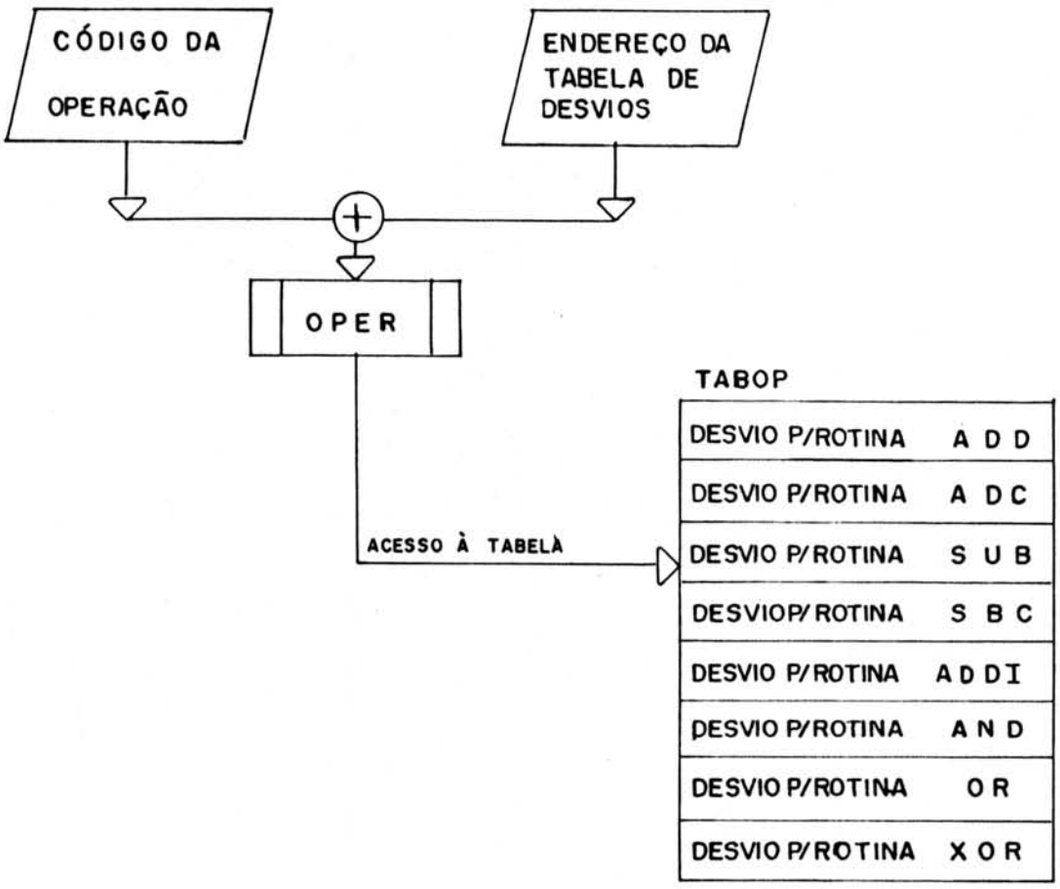
3.4.3.6 Acesso a Registrador

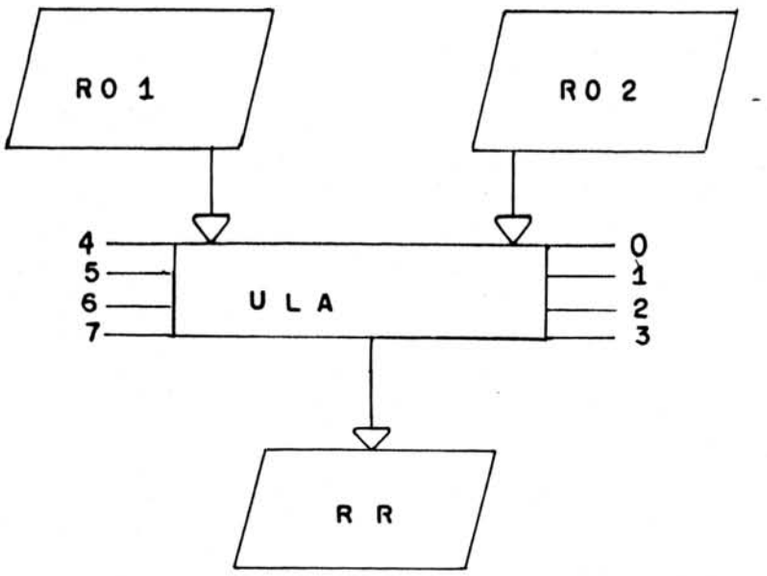


NOTAÇÃO:

Rn - Registrador a ser acessado

3.4.3.7 Operações da ULA





NOTAÇÃO:

R01 - Registrador que contém operando 1

R02 - Registrador que contém operando 2
RR - Registrador que contém resultado

0\ ADD: RR <-- R01 + R02

1\ ADC: RR <-- R01 + R02 + C

2\ SUB: RR <-- R01 - R02

3\ SBC: RR <-- R01 - R02 - C

4\ ADI: RR <-- R01 + SHR (R02)

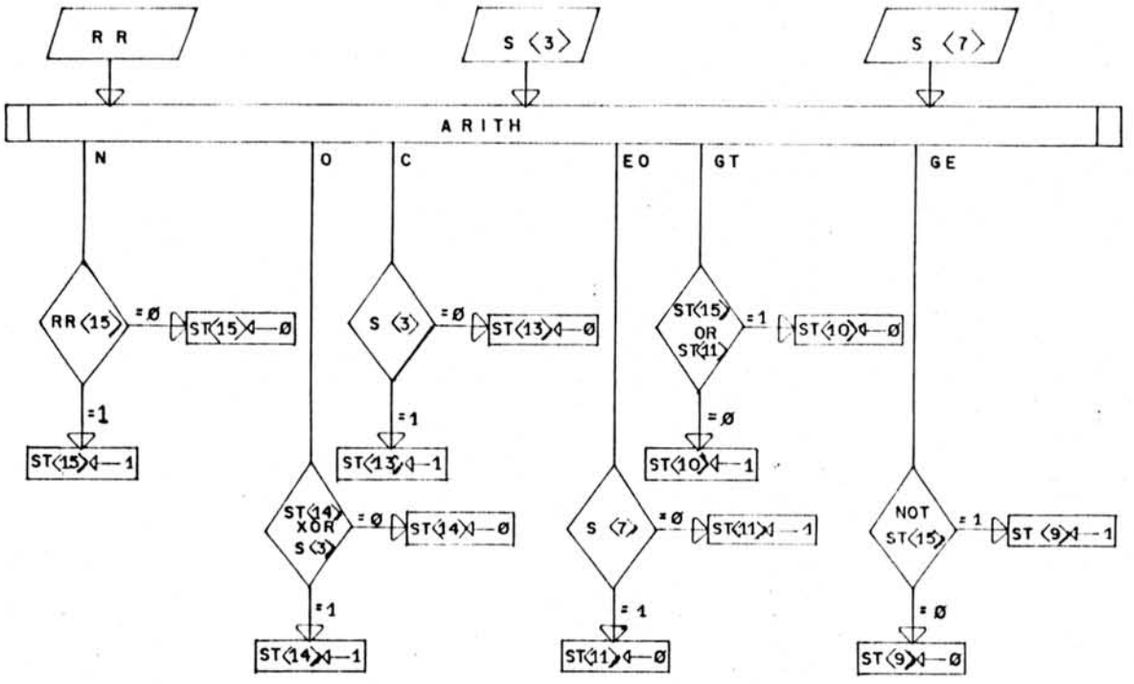
LB <-- R02 <0>

5\ AND: RR <-- R01 and R02

6\ OR: RR <-- R01 or R02

7\ XOR: RR <-- R01 xor R02

3.4.3.8 Atualização do Registrador de Estados do Processador



NOTAÇÃO:

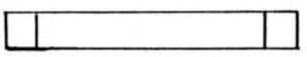
RR - registrador que contém o resultado de uma operação da ULA

S - registrador de estados do ED 311

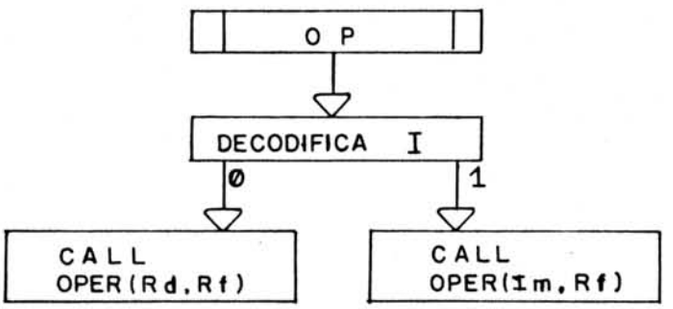
3.4.4 Rotinas das Instruções

Estas são as rotinas principais do emulador. As chamadas às rotinas auxiliares são representadas pela instrução "CALL".

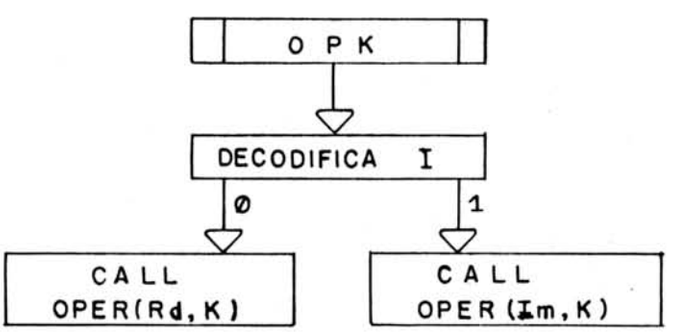
Para facilidade de consulta, os nomes das rotinas são os mesmos utilizados no programa, usando o símbolo abaixo.



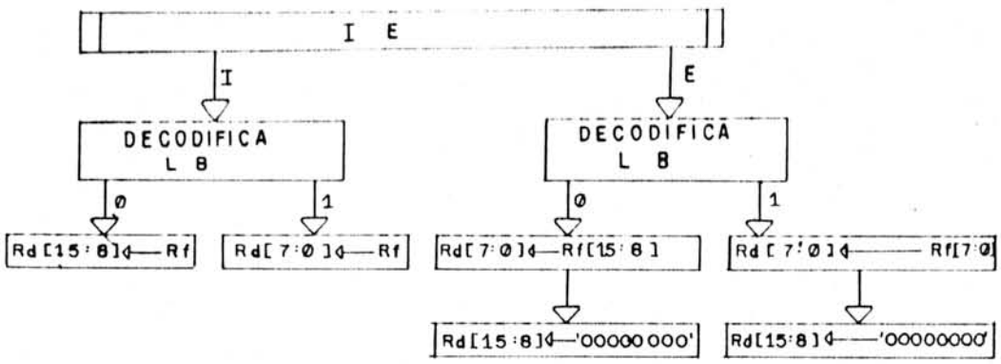
3.4.4.1 Instrução OP



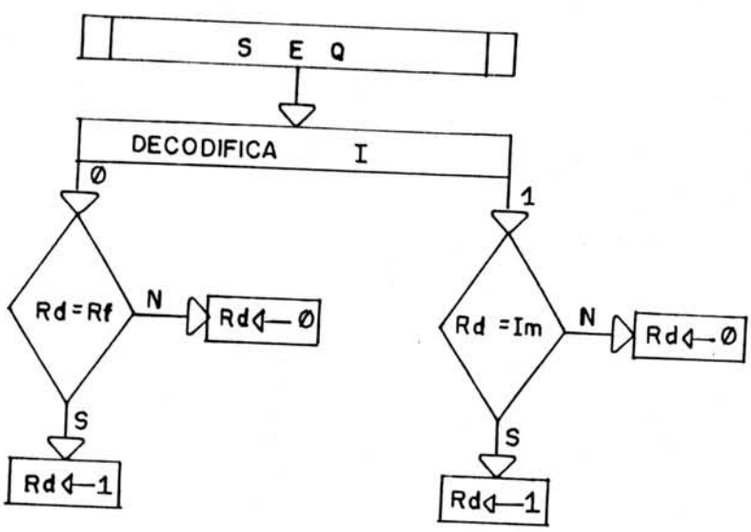
3.4.4.2 Instrução OPK



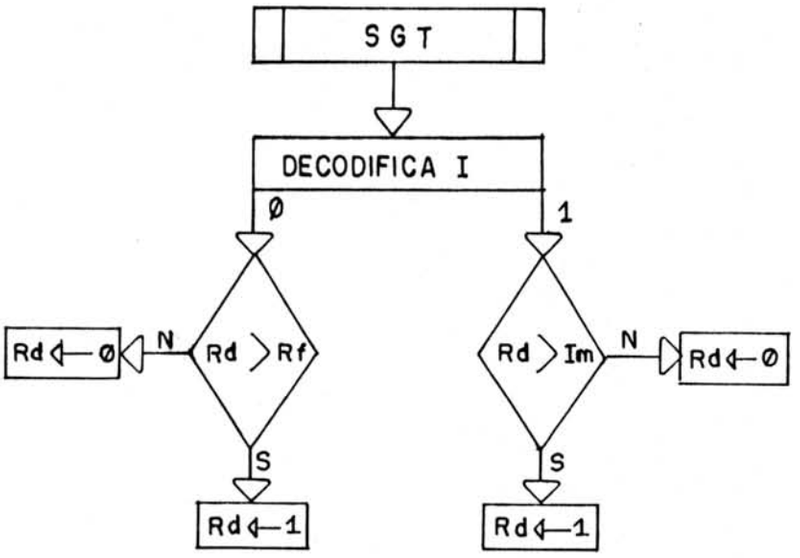
3.4.4.3 Instrução IE



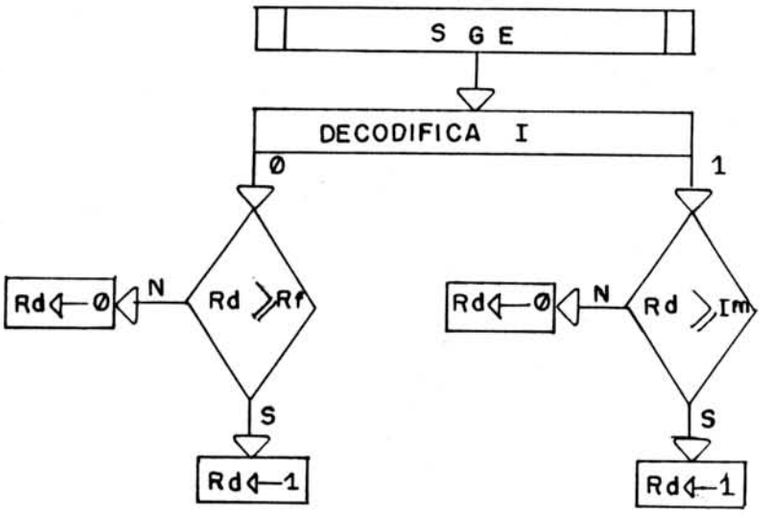
3.4.4.4 Instrução SEQ



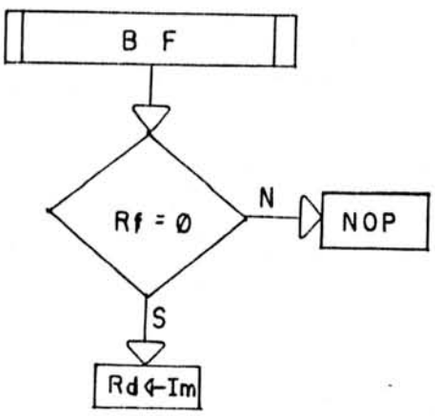
3.4.4.5 Instrução SGT



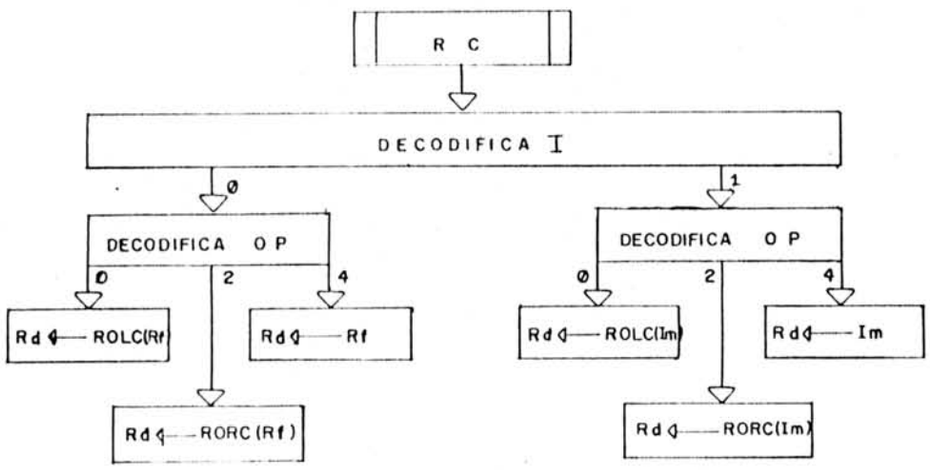
3.4.4.6 Instrução SGE



3.4.4.7 Rotina BF



3.4.4.8 Rotina RC

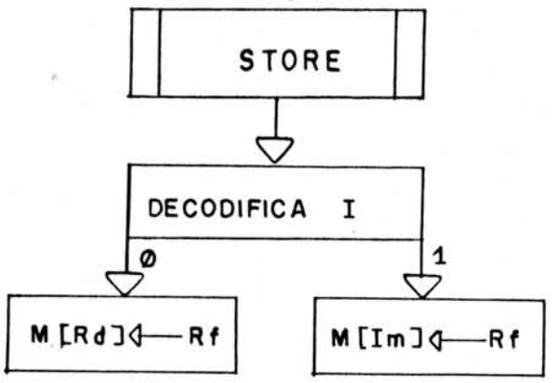


NOTAÇÃO:

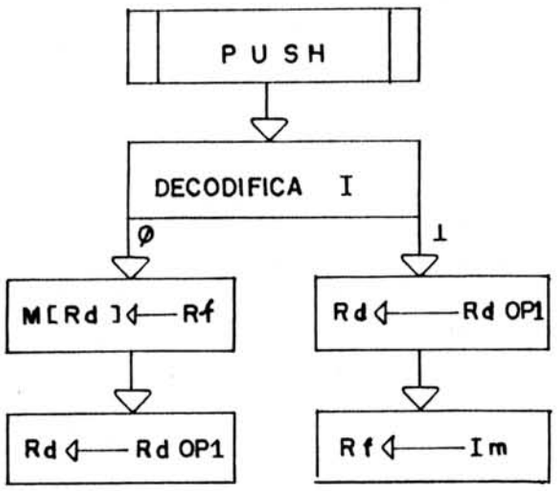
ROLC - rotação à esquerda com carry

RORC - rotação à direita com carry

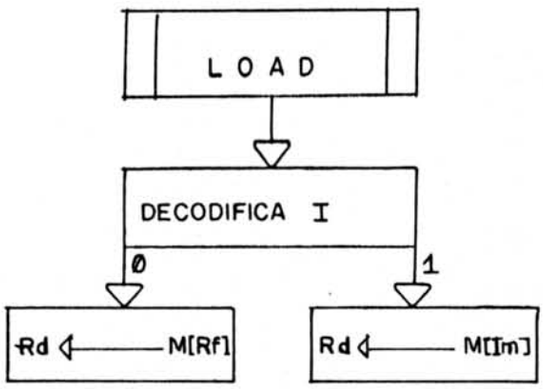
3.4.4.9 Instrução STORE



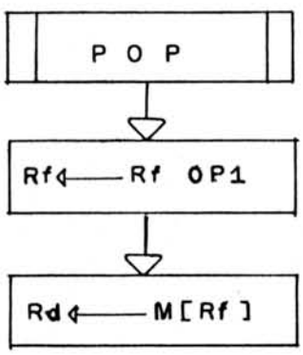
3.4.4.10 Instrução PUSH



3.4.4.11 Instrução LOAD



3.4.4.12 Instrução POP



4 MONITORAÇÃO DAS INSTRUÇÕES DO PCIR

Este capítulo apresenta as características do monitor de instruções da máquina PCIR.

As instruções são monitoradas em tempo de execução e contabilizadas conforme o tipo da instrução para produzir a estatística do programa executado.

Os procedimentos que realizam as tarefas mencionadas estão incluídos no emulador mas há uma participação do montador de programas PCIR e do gerente de módulos objeto para a máquina PCIR que são apresentados nos próximos capítulos.

4.1 Processo de Monitoração

As instruções utilizadas pelo programador PCIR podem ser monitoradas pelo emulador objetivando contabilizar o uso de cada tipo de instrução. Isto é feito por um monitor, embutido no emulador, que pode ser acionado juntamente com a máquina emulada.

O emulador mantém um vetor de 31 palavras na memória de controle que é manipulado conforme o algoritmo apresentado na seção 4.2.

Há ainda uma rotina de nome "ESTATI" também embutida no emulador e invocada pelo gerente de módulos objeto que está descrito no capítulo 5. Esta rotina é responsável pela conversão do valor do contador para a seqüência de caracteres que o representam e pela impressão da estatística do módulo.

Para indicar a opção de monitoração o usuário deverá especificar no programa fonte a diretiva especial "$MONITOR" descrita nos capítulos 6 e 7. Esta diretiva é reconhecida pelo montador o qual envia esta informação para o gerente de módulos objeto.

Caso não seja desejada a monitoração para não comprometer o tempo de execução das instruções, isto fica evidenciado com a ausência desta diretiva no programa fonte.

4.2 Algoritmo de Monitoração e Contabilização das Instruções

* IM - indica o formato da instrução
* CODINST - código da instrução
* CONST - constante (0, 1, -1)
* OPER - operação
* As variáveis utilizadas como contadores conservam os nomes usados no programa emulador (anexo 4).

```
inicio
  CTINST <-- CTINST + 1;
  decodifique IM;
  se IM = 1 então CTIM <-- CTIM + 1
            senão CTNIM <-- CTNIM + 1;
  decodifique CODINST;
  caso CODINST =
    OP, OPK : inicio
                decodifique OPER;
                caso OPER =
                  ADD  : CTADD <-- CTADD + 1;
                  ADDC : CTADDC <-- CTADDC + 1;
                  ADDI : CTADI <-- CTADI + 1;
                  SUB  : CTSUB <-- CTSUB + 1;
                  SUBC : CTSBC <-- CTSBC + 1;
                  AND  : CTAND <-- CTAND + 1;
                  OR   : CTOR  <-- CTOR + 1;
                  XOR  : CTXOR <-- CTXOR + 1;
                fim;
```

```
                se CODINST = OPK
                   então inicio
                           decodifique CONST;
                           caso CONST =
                             0 : CTKZER <-- CTKZER + 1;
                             1 : CTKUM <-- CTKUM + 1;
                            -1 : CTKUMN <-- CTKUMN + 1;
                           fim
                         fim;
              fim;
     IE     : inicio
              CTIE <-- CTIE + 1;
              decodifique OPER;
              se OPER = INSB então CTINSB <-- CTINSB + 1
                                  senão CTEXTB <-- CTEXTB + 1;
            fim;
     SEQ    : CTSEQ <-- CTSEQ + 1;
     SGT    : CTSGT <-- CTSGT + 1;
     SGE    : CTSGE <-- CTSGE + 1;
     BF     : CTBF  <-- CTBF + 1;
     RC     : inicio
              CTRC <-- CTRC + 1;
              decodifique OPER;
              caso OPER =
                ROLC : CTROLC <-- CTROLC + 1;
                RORC : CTRORC <-- CTRORC + 1;
                MOVE : CTMOVE <-- CTMOVE + 1;
            fim;
     STORE  : CTSTOR <-- CTSTOR + 1;
     PUSH   : CTPUSH <-- CTPUSH + 1;
     LOAD   : CTLOAD <-- CTLOAD + 1;
     POP    : CTPOP  <-- CTPOP + 1;
fim.
```

5 GERENTE DE MÓDULOS OBJETO PARA A MÁQUINA PCIR

Este capítulo descreve o ambiente do usuário PCIR enquanto conviver com a máquina emulada.

5.1 Definição

O gerente de módulos objeto consiste de uma camada externa à máquina que tem a função básica de gerir a carga e execução dos programas PCIR.

A necessidade imediata deste gerente para testar o próprio software em projeto, não permitiu maiores sofisticações. Constituindo-se portanto de uma ferramenta rudimentar que visa livrar o usuário do ambiente pouco convidativo do ED 311 e de alguns detalhes de implementação.

Procurou-se ainda automatizar algumas tarefas que se fazem necessárias quando se interage com uma máquina "nua".

5.2 Implementação

O gerente de módulos objeto para a máquina PCIR foi escrito na linguagem de microprogramação descrita no anexo 1 e foi incorporado ao emulador.

Utiliza-se das rotinas já descritas para o emulador e de rotinas disponíveis no MDE.

5.2.1 Características Funcionais

Devido à pouca disponibilidade de memória na máquina PCIR enquanto emulada, os programas maiores poderão ser divididos em módulos independentes e carregados e executados com uma interferência externa mínima, qual seja, acionar uma tecla avisando que a carga ou execução pode prosseguir. Esta interação objetiva oferecer ao usuário a possibilidade de trocar de disquete antes de carregar ou executar o próximo módulo, enquanto o gerente mantém a máquina ou o carregador em estado de espera.

Os módulos podem ser sobrepostos na mesma área de armazenamento e poderão compartilhar uma "pseudo-área de dados globais" conforme exemplificado na seção 5.2.1.1, não podendo entretanto, compartilhar código.

Quando a máquina PCIR é "ligada", o controle é imediatamente passado ao gerente o qual inicialmente aciona o carregador (seção 5.2.4) para carregar o primeiro setor do disquete onde foram gravadas pelo montador (capítulo 6) as informações relativas à carga e execução dos módulos a serem geridos. Estas informações guiarão o gerente para que carregue um módulo, ative sua execução, carregue o módulo seguinte, ative sua execução e assim sucessivamente até encontrar a informação de que não existem mais módulos para serem executados.

Após a execução de cada módulo, o gerente verifica se a opção de monitoração está ativa e aciona a rotina "ESTATI" (seção 4.1) responsável pela impressão da contabilização das instruções monitoradas.

5.2.1.1 Pseudo-área de Dados Globais

O compartilhamento de dados entre os diferentes módulos pode ser conseguido, usando o artifício mostrado na figura 5.1 onde supõe-se que dois módulos A e B quaisquer desejam compartilhar uma pilha de 100 posições.

Os endereços destas áreas de dados devem ser cuidadosamente escolhidos pois não há proteção contra invasões indevidas nestas áreas.

módulo A			módulo B		
X	EQU	*	Y	EQU	*
	ORG	H'FF0		ORG	H'FF0
GPILHAA	RES	100	GPILHAB	RES	100
	ORG	X		ORG	Y

FIGURA 5.1 Compartilhamento de dados entre os módulos A e B

5.2.2 Configuração do Disquete

O disquete visto pelo gerente apresenta a formatação descrita abaixo, usando as estruturas de dados conhecidas da linguagem Pascal:

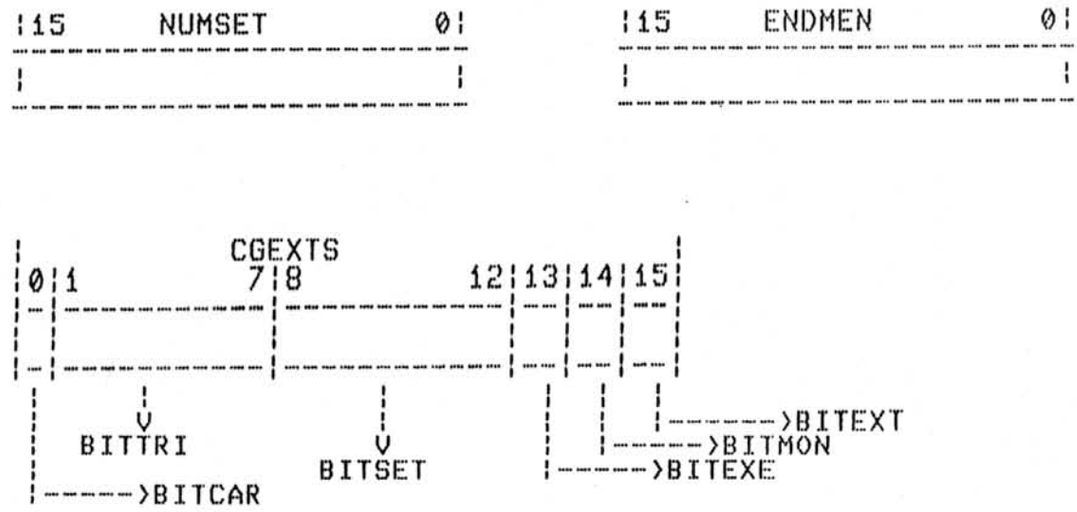
```
DISQUETE = array [0:76] of TRILHA
TRILHA = array [1:26] of SETOR
SETOR = array [1:64] of WORD
WORD = array [0:15] of BITS
BITS= (0,1)
```

5.2.2.1 Configuração da Área de Carga

O primeiro setor da trilha zero do disquete está reservado para informações relativas à carga dos módulos. Este setor é então visto e tratado pelo gerente, de forma especial em relação aos demais .

A formatação do setor 1 está representada na estrutura de dados abaixo é e ilustrada na figura 5.2.

```
SETOR1 = array [1:21] of INFOCAR
INFOCAR = registro
          CGEXTS : WORD
          NUMSET : WORD
          ENDMEN : WORD
```



NUMSET - palavra que contém o número de setores a serem carregados.

ENDMEN - palavra que contém o endereço de carga na memória principal.

CGEXTS - palavra que contém os seguintes campos:

BITCAR - ativação de carga
BITTRI - trilha inicial do módulo objeto
BITSET - setor inicial do módulo objeto
BITEXE - ativação de execução
BITMON - opção de monitoração
BITEXT - indica inicio de módulo externo

FIGURA 5.2 Formatação do Setor 1

5.2.3 Algoritmo do Gerente de módulos Objeto

```
 * CONTRI - contador de registros INFOCAR
 * BCG,  BEX,  BMO,  BEXT, CARTRI, CARSET, CARNUM, CAREND -
   recebem o conteúdo dos registros INFOCAR
 * AREACG - área de carga
 * PCINI - parâmetro para o emulador
inicio
  EXIBEVIDEO ("CARGA?");
  AREACG <---DISQUETE [0].TRILHA [1];
  enquanto houver módulos faça
      inicio
        repetir

            CONTRI <-- CONTRI + 1;
            BCG <-- AREACG [CONTRI].CGEXTS.BITCAR;
            BEX <-- AREACG [CONTRI].CGEXTS.BITEXE;
            BEXT <-- AREACG [CONTRI].CGEXTS.BITEXT;
            BMO <-- AREACG [CONTRI].CGEXTS.BITMON;
            CARTRI <-- AREACG [CONTRI].CGEXTS.BITTRI;
            CARSET <-- AREACG [CONTRI].CGEXTS.BITSET;
            CARNUM <-- AREACG [CONTRI].NUMSET;
            CAREND <-- AREACG [CONTRI].ENDMEN;
            se BEXT = 1 então PCINI <-- CAREND;
            CARREGADOR (CARTRI, CARSET, CARNUM, CAREND);
        ate BEX <> 0;
        EXIBEVIDEO ("CARPT");
        EXIBEVIDEO ("EX?");
        EMULADOR (PCINI);
        EXIBEVIDEO ("EXPT");
        se BMO = 1 então ESTATI;
        EXIBEVIDEO ("CARGA?")
      fim; (enquanto)
fim.
```

5.2.4 Carregador

O carregador de módulos objeto quando invocado pelo gerente recebe os parâmetros necessários à carga de uma seqüência de palavras do disquete para a memória principal, já que os módulos vistos pelo carregador são seqüências de código de máquina com o respectivo endereço de carga.

Para o usuário, cada módulo corresponde a um programa e esta visão externa é mantida pelo gerente o qual aciona automáticamente o carregador a cada segmento a ser carregado e interage com o usuário somente quando todos os segmentos de um módulo já estão carregados. Denominamos segmento de um módulo, cada trecho do programa objeto que exige um novo endereço de carga, ou seja, a cada instrução ORG de um programa fonte, conforme descrito na seção 7.3.2.

O carregador utiliza a rotina FDRD do MDE e realiza a carga conforme o algoritmo apresentado na próxima seção.

5.2.4.1 Algoritmo do Carregador

```
CARREGADOR (CARTRI, CARSET, CARNUM, CAREND)
inicio
  TRILHA <-- CARTRI;
  SETOR <-- CARSET;
  ENDER <-- CAREND;
  CONT <-- CARNUM;
  repetir
      MDE.FDRD (SETOR, ENDER);
      SETOR <-- SETOR + 1;
      se SETOR = 27
         então inicio
                 TRILHA <-- TRILHA + 1;
                 SETOR <-- 1;
                 ENDER <-- ENDER + 128;
                 CONT <-- CONT - 1
               fim;
  até CONT = 0;
fim.
```

5.2.5 Comunicação com o Usuário

A comunicação entre o gerente e o usuário é feita através do procedimento EXIBEVIDEO, que quando invocado pede permissão, através de um ponto de interrogação para prosseguir com a carga ou execução do próximo módulo e informa através dos caracteres PT, o término da carga ou execução do módulo anterior. O acionamento de uma tecla pelo usuário indica que o módulo está pronto para ser operado.

Este procedimento utiliza as rotinas CPL de comunicação com os periféricos lentos disponíveis no MDE. O

algoritmo de comunicação é apresentado abaixo:

```
inicio
  EXIBEVIDEO ("mensagem");
  enquanto não permite faça
            MDE.CPL ("mensagem");
fim
```

Observações:

* retorna ao gerente após acionada uma tecla
* "mensagem" significa:

CARGA?	antes de carregar o módulo
CARPT	após efetuada a carga
EX?	antes de executar o módulo
EXPT	após a execução do módulo

6 MONTADOR PCIR

Este capítulo descreve as instruções do montador PCIR e as estruturas utilizadas na sua realização.

6.1 Introdução

O montador PCIR é um programa tradutor de duas passagens que recebe como entrada um programa em linguagem de montagem PCIR e produz como saída um programa em linguagem de máquina PCIR.

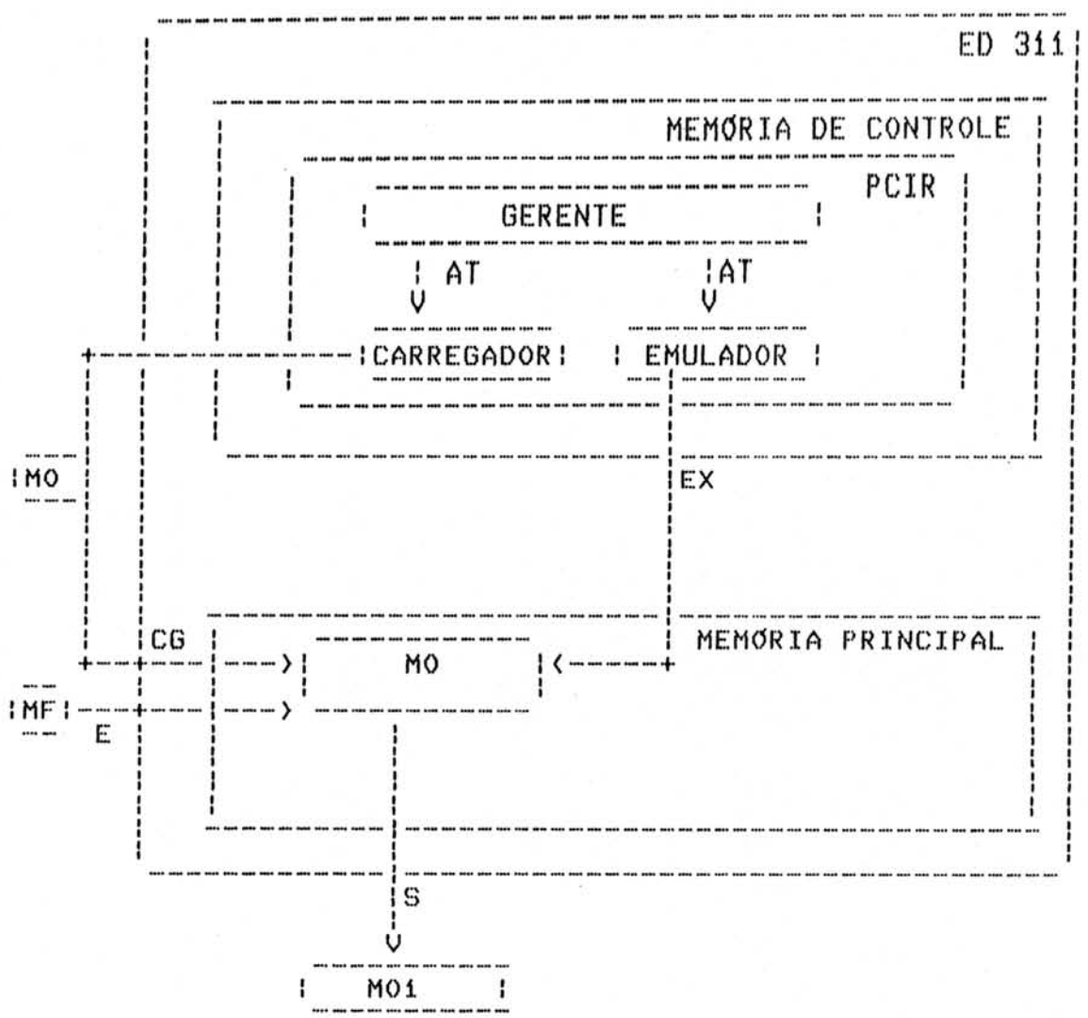
Para possibilitar a migração deste montador, da máquina hospedeira para a máquina alvo, ele foi escrito na própria linguagem de montagem do PCIR e foi montado pela primeira vez por um montador cruzado disponível no microcomputador ED 251 /BOR 87/.

Após este processo inicial, o montador é carregado no ED 311 e é validado montando a si próprio. A partir desta fase, entra em execução normal para tradução de programas fonte PCIR.

As figuras 6.1, 6.2 e 6.3 ilustram as três fases descritas acima.

MF - MONTADOR FONTE (linguagem de montagem PCIR)
MC - MONTADOR CRUZADO
MO - MONTADOR OBJETO (linguagem de máquina PCIR)

FIGURA 6.1 Primeira Fase: Processo Inicial



AT - ATIVAÇÃO

CG - CARGA

EX - EXECUÇÃO

E - ENTRADA

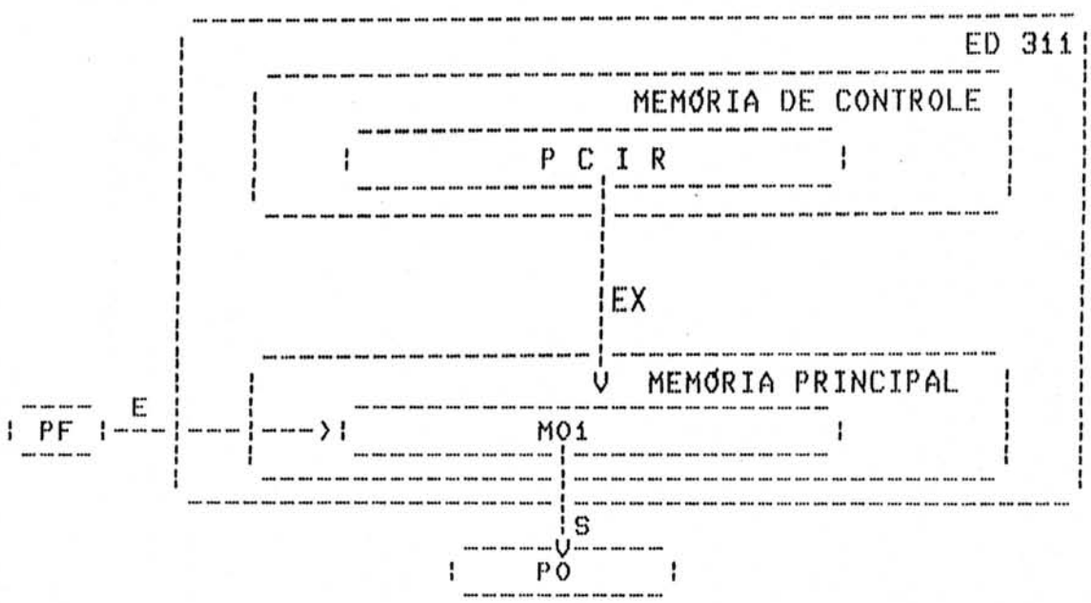
S - SAÍDA

MF - MONTADOR FONTE

MO - MONTADOR OBJETO (produzido no processo inicial)

MO1 - MONTADOR OBJETO VÁLIDO (produzido nesta fase)

FIGURA 6.2 Segunda Fase: Validação do Montador PCIR



E - ENTRADA
S - SAÍDA
EX - EXECUÇÃO
MO1 - MONTADOR OBJETO PRODUZIDO NA FASE ANTERIOR
PF - PROGRAMA FONTE PCIR (programa do usuário)
PO - PROGRAMA OBJETO PCIR (executável pela máquina PCIR)

FIGURA 6.3 Terceira Fase: Execução Normal

6.2 Arquitetura Visível ao Programador

O capítulo 2 apresentou a estrutura do microprocessador PCIR, sem preocupação com o usuário. Aqui mostra-se a máquina PCIR sob o ponto de vista do programador.

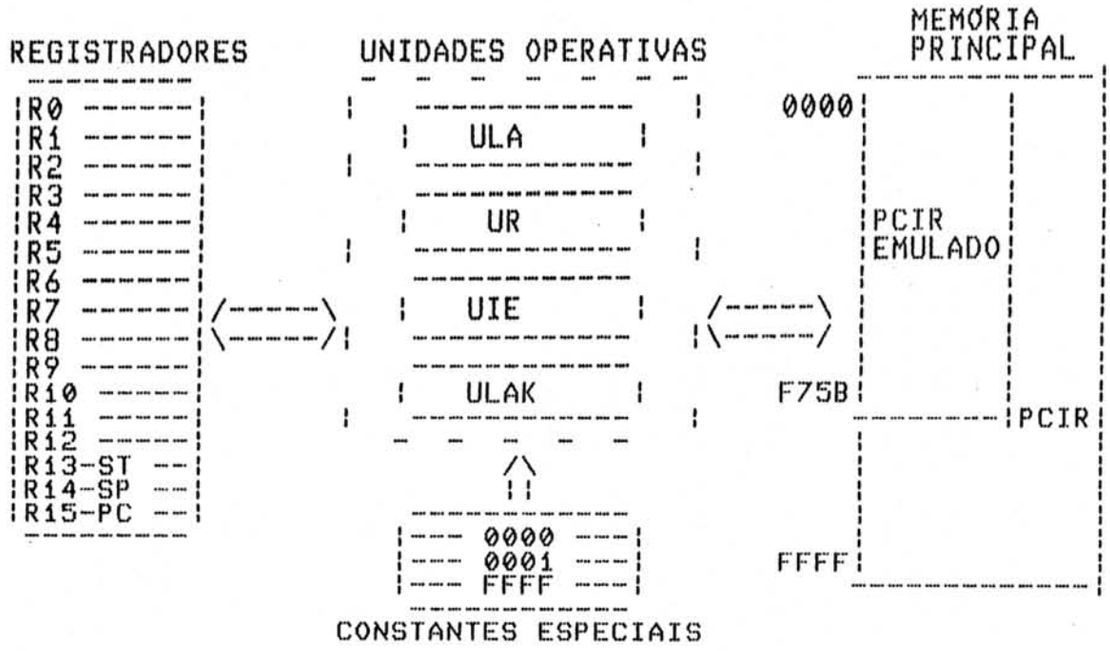
O microprocessador PCIR apresenta um conjunto de 16 registradores de 16 bits que serão referidos neste trabalho usando a nomenclatura que aparece na figura 6.4.

Destes registradores, 13 são de propósito geral (R0 a R12) e 3 são de uso específico (R13 a R15) e por isso recebem uma denominação especial, conforme figura 6.4.

Todos os registradores podem ser especificados como fonte ou destino das instruções realizadas pelas unidades operativas mostradas na figura 6.4.

As instruções realizadas pela ULAK usam além dos registradores, as constantes especiais (0, 1,-1) mostradas em hexadecimal na figura 6.4.

A memória principal é acessada por palavra de 16 bits nos limites especificados em hexadecimal na figura 6.4.



ULA - UNIDADE ARITMÉTICA E LÓGICA

UR - UNIDADE DE ROTAÇÃO

UIE - UNIDADE DE INSERÇÃO E EXTRAÇÃO DE BYTES

ULAK - UNIDADE ARITMÉTICA E LÓGICA COM CONSTANTES

PC - PONTEIRO DE INSTRUÇÕES (*)

SP - PONTEIRO DE PILHA

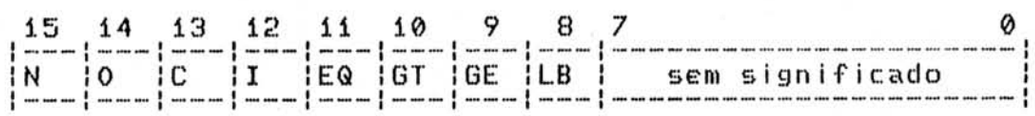
ST - REGISTRADOR DE ESTADOS DO PROCESSADOR

(*) - Devido à busca antecipada de instruções, a modificação do PC fica retardada de uma instrução

FIGURA 6.4 Arquitetura Visível ao Programador

6.2.1 Registrador de Estados do Processador

O registrador de estados do processador (R15) exige uma descrição mais detalhada em relação ao significado de cada um dos seus bits. A figura 6.5 mostra os bits que são usados para indicar os diferentes estados.



N - indica resultado negativo de uma operação aritmética e lógica

O - indica estouro de capacidade na realização de uma operação aritmética e lógica (OVERFLOW)

C - indica "vai-um" (CARRY) do bit 15 em uma operação aritmética e lógica e recebe o bit de rotação em uma operação de transferência de dados

I - indica interrupção habilitada

EQ - indica resultado igual a zero de uma operação aritmética e lógica

GE - indica resultado maior ou igual a zero em uma operação aritmética e lógica

LB - indica o local de byte para uma operação de transferência de bytes e recebe o bit deslocado de uma operação de adição para índice

FIGURA 6.5 Configuração do Registrador de Estados do Processador

6.3 Descrição dos Tipos de Dados

As próximas seções descrevem os tipos de dados reconhecidos pelo montador PCIR, que podem ser utilizados na representação dos operandos.

6.3.1 Nome Simbólico

Nome simbólico é uma seqüência de até oito caracteres (letras maiúsculas e dígitos), sendo o primeiro caractere obrigatoriamente alfabético.

Exemplos:

a) MONTADOR
b) A12
c) PILHA

6.3.2 Constantes

Uma constante é um valor numérico que pode ser representado na base decimal, na base hexadecimal, ou como caractere.

Uma constante decimal consiste de um conjunto de até 5 dígitos que pode ser precedido do sinal mais (+) ou menos (-).

Uma constante hexadecimal consiste de um conjunto de no máximo 4 caracteres hexadecimais e deve ser precedido pelo prefixo H'.

Uma constante caractere consiste de no máximo dois caracteres ASCII precedidos pelo prefixo C`.

Exemplos:

```
a) H`35FA      ;constante hexadecimal
b)   1236      ;constante decimal positiva
c) +1236       ;constante decimal positiva
d) -1236       ;constante decimal negativa
```

6.3.3 Cadeia de Caracteres

Uma cadeia de caracteres consiste de um conjunto de até 8 caracteres ASCII que deve ser precedido do prefixo C`.

Exemplos:

```
a) C`montador
b) C`10A23
c) C`ADD
```

6.3.4 Referência a Contador de Posição

A referência a o contador de posição é especificada através do símbolo especial * que deve ser usado para se referir ao endereço da primeira posição de memória da instrução que usa o símbolo.

6.3.5 Expressão

Expressão é uma combinação aritmética (através dos operadores + e -) de nome simbólico, constante decimal ou hexadecimal e referência a contador de posição que podem ser incluídas dentro de parênteses indicando agrupamento de operações.

Exemplos:

a) A12 + PILHA
b) A12 - (H`35FA + 1236 - PILHA)
C) * -1

6.4 Definição WSN dos Tipos de Dados

Devido às muitas variações da notação BNF (BACKUS-NAUR FORM) comumente usada /CAL 79/, adotou-se a notação recomendada por Wirth /WIR 77/, e denominada WSN (WIRTH SYNTAX NOTATION), onde é usada a seguinte convenção:

$\{\ \}_{0}^{n}$ - indica a repetição do ítem, onde n representa o número máximo de repetições

! - indica alternância

" " - indica símbolo terminal

( ) - indica agrupamento de símbolos

!...! - indica a continuação da série (não faz parte do WSN)

6.4.1 Tipos de Dados (WSN)

EXPRESSÃO = TERMO ¦ EXPRESSÃO ("+" ¦ "-") TERMO.

TERMO = SÍMBOLO ¦ CONSTANTE ¦ "*" ¦ "(" EXPRESSÃO ")".

CONSTANTE = CONSTANTE DECIMAL ¦ CONSTANTE HEXADECIMAL.

SÍMBOLO = LETRA (LETDIG)$_{0}^{7}$.

CONSTANTE DECIMAL = ("+" ¦ "-")$_{0}^{1}$ (DÍGITO)$_{1}^{5}$.

CONSTANTE HEXADECIMAL = "H" (DIGHEXA)$_{1}^{4}$

CADEIA DE CARACTERES = C (LETDIGESP)$_{1}^{8}$.

LETDIG = LETRA ¦ DÍGITO.

LETDIGESP = LETDIG ¦ ESPECIAL.

LETRA = "A" ¦ "B" ¦.....¦ "Z".

DÍGITO = "0" ¦ "1" ¦.....¦"9".

DIGHEXA = DIGITO ¦ "A" ¦ "B" ¦ "C" ¦ "D" ¦ "E" ¦ "F".

ESPECIAL = "b" ¦ "!" ¦...¦ "/" ¦ ";" ¦...¦ "@" ¦...¦"."

6.5 Conjunto de Instruções

A partir do conjunto de instruções do microprocessador PCIR, definiu-se a linguagem de montagem PCIR. Os mnemônicos usados nesta linguagem seguem a proposição de "Uma Linguagem de Montagem Padrão para os Microprocessadores", definida em /FIS 79/.

As instruções em linguagem de montagem foram definidas visando a simplicidade do tradutor e procurando conservar a potência das instruções da máquina para que,

dessa forma, este conjunto de instruções possa servir de linguagem alvo de um tradutor de mais alto nível, sem perda de eficiência.

Estas instruções estão divididas em quatro grupos, quais sejam:

a) Instruções de Máquina - São instruções que possuem um código de máquina correspondente.

b) Diretivas de Montagem - São instruções que permitem ao usuário reservar espaço de armazenamento, determinar endereço de carga e definir sinônimos.

c) Diretivas de Listagem - São instruções que oferecem ao usuário opções quanto às listagens produzidas pelo montador.

d) Diretivas Especiais - São instruções que permitem ao usuário incluir rotinas de entrada e saída em seu programa e monitorar a execução das instruções de máquina.

6.5.1 Instruções de Máquina

As instruções de máquina estão divididas conforme a função em:

a) Instruções Aritméticas e Lógicas

b) Instruções de Transferência de Bytes

c) Instruções de Transferência de Dados

d) Instruções de Comparação

e) Instruções de Desvio

f) Instruções de Transferência entre Registradores e Memória

g) Instruções Diversas

Cada grupo de instruções dos tipos mencionados acima são apresentadas na forma de tabelas nas próximas seções onde sob o título "INSTRUÇÃO SIMBÓLICA", aparece o mnemônico da instrução com os operandos, os quais podem ser de um dos seguintes tipos:

a) **Registrador** - Representado por Rf ou Rd, conforme a função (fonte ou destino), significando qualquer registrador de 16 bits consistente com a arquitetura apresentada na seção 6.2.

Os registradores devem ser especificados através de constante decimal, constante hexadecimal, nome simbólico ou expressão.

b) **Constante Especial** - Representada por Rk, antecedida do símbolo especial #, significando uma das constantes consistente com a arquitetura apresentada na seção 6.2.

Estas constantes especiais devem ser especificadas através de constante decimal indicando o próprio valor.

c) **Imediato** - Representado por Im antecedido do símbolo especial =, significando um valor de 16 bits.

Os valores imediatos devem ser especificados

através de constante decimal, constante hexadecimal, nome simbólico ou expressão.

d) **Símbolo** - Representado por ROTULO, significando um valor de 16 bits.

O símbolo deve ser especificado através de um nome simbólico.

Se a coluna "ESTADOS" estiver presente na tabela, um X indica os bits do registrador de estados do processador que são modificados na execução da instrução. A ausência desta coluna significa que aquele grupo de instruções não afeta o registrador de estados.

Sob o título "SIGNIFICADO", são especificadas as operações realizadas na execução das instruções, usando as seguintes convenções:

M[ ]	- indica conteúdo de memória
OP( )	- indica aplicação da operação OP
+	- indica soma aritmética
-	- indica subtração aritmética
<--	- indica atribuição
/\	- indica E lógico
\/	- indica OU lógico
\/	- indica OU exclusivo lógico
<p:q>	- indica bits de "p" a "q"
C	- indica "vai-um" do bit 15 (carry)
O	- indica overflow
N	- indica resultado negativo
EQ	- indica resultado igual a zero
GT	- indica resultado maior que zero
GE	- indica resultado maior ou igual a zero

LB - indica localização do byte na palavra

Z - indica resultado zero

SHR - indica operação de deslocamento à direita

SHL - indica operação de deslocamento à esquerda

6.5.1.1 Instruções Aritméticas e Lógicas

As instruções aritméticas e lógicas são realizadas pela ULA e ULAK. Consistem de operações aritméticas e lógicas entre dois registradores, entre um valor imediato e um registrador, entre um registrador e uma constante especial e entre um valor imediato e uma constante especial.

As instruções de adição e subtração podem ainda operar com o bit de "vai-um" (carry) do registrador de estados do processador.

O grupo de instruções aritméticas e lógicas está listado na tabela 6.1, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.1 Instruções Aritméticas e Lógicas

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO	ESTADOS						
MNEMÔNICO	OPERANDOS		N	O	C	EQ	GT	GE	LB
		SOMA ARITMÉTICA							
ADD	Rd,Rf	Rd ⟨--Rd + Rf	x	x	x	x	x	x	
ADD	Rd,Rf,=Im	Rd ⟨--Im + Rf	x	x	x	x	x	x	
ADD	Rd,#Rk	Rd ⟨--Rd + Rk	x	x	x	x	x	x	
ADD	Rd,#Rk,=Im	Rd ⟨--Im + Rk	x	x	x	x	x	x	
		SOMA COM CARRY							
ADDC	Rd,Rf	Rd ⟨--Rd + Rf + C	x	x	x	x	x	x	
ADDC	Rd,Rf,=Im	Rd ⟨--Im + Rf + C	x	x	x	x	x	x	
ADDC	Rd,#Rk	Rd ⟨--Rd + Rk + C	x	x	x	x	x	x	
ADDC	Rd,#Rk,=Im	Rd ⟨--Im + Rk + C	x	x	x	x	x	x	
		SUBTRAÇÃO ARITMÉTICA							
SUB	Rd,Rf	Rd ⟨--Rd - Rf	x	x	x	x	x	x	
SUB	Rd,Rf,=Im	Rd ⟨--Im - Rf	x	x	x	x	x	x	
SUB	Rd,#Rk	Rd ⟨--Rd - Rk	x	x	x	x	x	x	
SUB	Rd,#Rk,Im	Rd ⟨--Im - Rk	x	x	x	x	x	x	
		SUBTRAÇÃO COM CARRY							
SUBC	Rd,Rf	Rd ⟨--Rd - Rf -C	x	x	x	x	x	x	
SUBC	Rd,Rf,=Im	Rd ⟨--Im - Rf - C	x	x	x	x	x	x	
SUBC	Rd,#Rk	Rd ⟨--Rd - Rk - C	x	x	x	x	x	x	
SUBC	Rd,#Rk,=Im	Rd ⟨--Im - Rk - C	x	x	x	x	x	x	
		SOMA DE ÍNDICE							
ADDI	Rd,Rf	Rd ⟨--Rd + SHR (Rf) LB ⟨--Rf⟨0⟩	x	x	x	x	x	x	x
ADDI	Rd,Rf,=Im	Rd ⟨--Im + SHR (Rf) LB ⟨--Rf⟨0⟩	x	x	x	x	x	x	x
ADDI	Rd,#Rk	Rd ⟨--Rd + SHR (Rk) LB ⟨--Rk⟨0⟩	x	x	x	x	x	x	x
ADDI	Rd,#Rk,=Im	Rd ⟨--Im + SHR (Rk) LB ⟨--Rk⟨0⟩	x	x	x	x	x	x	x

continua...

TABELA 6.1 Instruções Aritméticas e Lógicas (continuação)

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO	ESTADOS						
MNEMÔNICO	OPERANDOS		N	O	C	EQ	GT	GE	LB
		E LÓGICO							
AND	Rd,Rf	Rd <--Rd /\ Rf	x	x	x	x	x	x	
AND	Rd,Rf,=Im	Rd <--Im /\ Rf	x	x	x	x	x	x	
AND	Rd,#Rk	Rd <--Rd /\ Rk	x	x	x	x	x	x	
AND	Rd,#Rk,=Im	Rd <--Im /\ Rk	x	x	x	x	x	x	
		OU LÓGICO							
OR	Rd,Rf	Rd <--Rd \/ Rf	x	x	x	x	x	x	
OR	Rd,Rf,=Im	Rd <--Im \/ Rf	x	x	x	x	x	x	
OR	Rd,#Rk	Rd <--Rd \/ Rk	x	x	x	x	x	x	
OR	Rd,#Rk,=Im	Rd <--Im \/ Rk	x	x	x	x	x	x	
		OU EXCLUSIVO LÓGICO							
XOR	Rd,Rf	Rd <--Rd \/= Rf	x	x	x	x	x	x	
XOR	Rd,Rf,=Im	Rd <--Im \/= Rf	x	x	x	x	x	x	
XOR	Rd,#Rk	Rd <--Rd \/= Rk	x	x	x	x	x	x	
XOR	Rd,#Rk,=Im	Rd <--Im \/= Rk	x	x	x	x	x	x	
		INCREMENTO E DECREMENTO							
INC	Rd	Rd <--Rd + 1	x	x	x	x	x	x	
DEC	Rd	Rd <--Rd - 1	x	x	x	x	x	x	

6.5.1.2 Instruções de Transferência de Bytes

As instruções de transferência de bytes são realizadas pela "UIE", entre dois registradores e entre um valor imediato e um registrador. Consistem de operações de acesso aos bytes de uma palavra para extração ou inserção. O byte a ser acessado é designado pelo bit de "localização de byte" do registrador de estados do processador que é tornado 1 ou 0 pela instrução ADDI, descrita na seção

anterior.

O grupo de instruções de transferência de bytes está listado na tabela 6.2, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.2 Instruções de Transferência de Bytes

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO
MNEMÔNICO	OPERANDOS	
		INSERÇÃO DE BYTE
INSB	Rd,Rf	se ST(8) = 0 então Rd(7:0) <-- Rf(7:0) senão Rd(15:8) <-- Rf(7:0)
INSB	Rd,Rf,=Im	se ST(8) = 0 então Im(7:0) <-- Rf(7:0) Rd <-- Im senão Im(15:8) <-- Rf(7:0) Rd <-- Im
		EXTRAÇÃO DE BYTE
EXTB	Rd,Rf	se ST(8) = 0 então Rd(7:0) <-- Rf(7:0) Rd(15:8) <-- '0000' senão Rd(7:0) <-- Rf(15:8) Rd(15:8) <-- '0000'

6.5.1.3 Instruções de Transferência de Dados

As instruções de transferência de dados são realizadas pela UR entre dois registradores e entre um registrador e um valor imediato. Consistem de operações de transferência, sem acesso à memória, que podem realizar rotação com o bit "vai-um" (carry) do registrador de estados do processador antes de transferir o dado.

O grupo de instruções de transferência de dados está listado na tabela 6.3, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.3 Instruções de Transferência de Dados

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO	ESTADOS						
MNEMÔNICO	OPERANDOS		N	O	C	EQ	GT	GE	LB
		ROTAÇÃO À ESQUERDA COM CARRY							
ROLC	Rd,Rs	Rd ⟨-- SHL(Rf) Rd⟨0⟩ ⟨-- C C ⟨-- Rf⟨15⟩			x				
ROLC	Rd,=Im	Rd ⟨-- SHL(Im) Rd⟨0⟩ ⟨-- C C ⟨-- Im⟨15⟩			x				
		ROTAÇÃO À DIREITA COM CARRY							
RORC	Rd,Rf	Rd ⟨-- SHR(Rf) Rd⟨15⟩ ⟨-- C C ⟨-- Rf⟨0⟩			x				
RORC	Rd,=Im	Rd ⟨-- SHR(Im) Rd⟨15⟩ ⟨-- C C ⟨-- Im⟨0⟩			x				
		TRANSFERÊNCIA SEM ROTAÇÃO							
MOV	Rd,Rf	Rd ⟨-- Rf							
MOV	Rd,=Im	Rd ⟨-- Im							

6.5.1.4 Instruções de Comparação

As instruções de comparação são realizadas entre dois registradores e entre um registrador e um valor imediato. Consistem de operações relacionais cujo resultado é indicado pela atribuição dos valores -1 ou 0 ao registrador destino.

O grupo de instruções de comparação está listado na tabela 6.4, utilizando-se as definições e convenções

estabelecidas na seção 6.5.1.

TABELA 6.4 Instruções de Comparação

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO
MNEMÔNICO	OPERANDOS	
		COMPARA SE IGUAL
SEQ	Rd,Rf	se Rd = Rf então Rd <-- -1 senão Rd <-- 0
SEQ	Rd,=Im	se Rd = Im então Rd <-- -1 senão Rd <-- 0
		COMPARA SE MAIOR
SGT	Rd,Rf	se Rd > Rf então Rd <-- -1 senão Rd <-- 0
SGT	Rd,=Im	se Rd > Im então Rd <-- -1 senão Rd <-- 0
		COMPARA SE MAIOR OU IGUAL
SGE	Rd,Rf	se Rd >= Rf então Rd <-- -1 senão Rd <-- 0
SGE	Rd,=Im	se Rd >= Im então Rd <-- -1 senão Rd <-- 0

6.5.1.5 Instruções de Desvio

As instruções de desvio realizam desvio condicional, desvio incondicional, chamada de procedimento e retorno de procedimento.

O grupo de instruções de desvio está listado na tabela 6.5, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.5 Instruções de Desvio

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO
MNEMÔNICO	OPERANDOS	
BF	Rf,ROTULO	DESVIA SE FALSO se Rf = 0 então PC <-- rotulo
BR	ROTULO	DESVIO INCONDICIONAL PC <-- ROTULO
CALL	ROTULO	CHAMADA DE PROCEDIMENTO M[SP] <-- PC SP <-- SP - 1 PC <-- ROTULO
RET		RETORNO DE PROCEDIMENTO SP <-- SP + 1 PC <-- M[SP]

6.5.1.6 Instruções de Transferência entre Registradores e Memória

As instruções de transferência entre registradores e memória realizam a transferência de dados entre um registrador e uma posição de memória, nos dois sentidos.

O grupo de instruções de transferência entre registradores e memória está listado na tabela 6.6, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.6 Instruções de Transferência entre Registradores e Memória

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO
MNEMÔNICO	OPERANDOS	
		ARMAZENAMENTO
ST	Rd,Rf	M[RD] <-- Rf
ST	=IM,Rf	M[IM] <-- Rf
		CARGA
LD	Rd,Rf	Rd <-- M[Rf]
LD	Rd,=Im	Rd <-- M[Im]

6.5.1.7 Instruções Diversas

As instruções diversas realizam operações de manipulação de pilha e contém uma instrução não operante.

O grupo de instruções diversas está listado na tabela 6.7, utilizando-se as definições e convenções estabelecidas na seção 6.5.1.

TABELA 6.7 Instruções Diversas

INSTRUÇÃO SIMBÓLICA		SIGNIFICADO
MNEMÔNICO	OPERANDOS	
		EMPILHA
PUSH	Rd,Rf	M[Rd] <-- Rf Rd <-- Rd - 1
PUSH	Rd,Rf,=Im	M[Rd] <-- Rf Rd <-- Rd - 1 Rf <-- Im
		DESEMPILHA
POP	Rd,Rf	Rf <-- Rf + 1 Rd <-- M[Rf]
		NÃO OPERA
NOP		

6.5.2 Diretivas de Montagem

As diretivas de montagem são comandos para o montador, em vez de instruções para o microprocessador. Estes comandos indicam ao montador a execução de tarefas específicas durante o processo de montagem.

Na seção seguinte são apresentadas as diretivas de montagem do montador PCIR, usando as seguintes convenções:

< > - indica a obrigatoriedade do ítem

[ ] - indica a opcionalidade do ítem

( ) - indica tipos de dados alternativos

$(\ )_1^n$ - indica lista de tipos de dados alternativos separados por vírgula onde n representa o número máximo de repetições

Obs: O mnemônico da instrução aparece em letras maiúsculas.

6.5.2.1 Descrição e Sintaxe

a) Origem do Código : Atribui ao contador de posições vigente, o valor especificado pelo operando.

No início da montagem, o contador de posições é iniciado com zero. Um programa pode conter vários destes comandos e o montador não verifica se existe superposição de área. O formato do comando é mostrado abaixo.

```
                          ¦ nome simbólico ¦
[nome simbólico] ORG      ( expressão      )
                          ¦ constante      ¦
```

b) Definição de Sinônimo : Atribui um valor a um nome simbólico. O formato do comando é mostrado abaixo.

```
                          ¦ nome simbólico ¦
                          ¦ expressão      ¦
<nome simbólico> EQU      ( constante      )
                          ¦ *              ¦
                          ¦                ¦
```

c) Reserva de Memória : Reserva um bloco de palavras de memória. O conteúdo das posições de memória reservadas fica sem especificação. O formato do comando é mostrado abaixo.

```
                          ¦ nome simbólico ¦
[nome simbólico] RES      ( expressão      )
                          ¦ constante      ¦
                          ¦ *              ¦
```

d) **Reserva de Memória com Iniciação :** Inicia as próximas posições de memória com os valores especificados pelos operandos. O formato do comando é mostrado abaixo.

```
                                    n
                    ¦ constante     ¦
[nome simbólico] DATA ⟨              ⟩
                    ¦ cadeia de caracteres ¦
                                    1
```

e) **Término de programa:** Informa ao montador o fim do programa fonte. Deve ser o último comando do programa. O formato do comando é mostrado abaixo.

```
END
```

6.5.3 Diretivas de listagem

As diretivas de listagem são comandos que indicam ao módulo listador do montador a execução de tarefas específicas durante o processo de listagem:

a) **$HEAD ⟨título⟩**

Especifica o título que deve ser impresso em todas as listagens emitidas pelo montador e a cada troca de página provocada pela diretiva "$PAGE".

b) **$PAGE**

Esta diretiva faz com que o listador salte para uma nova página e imprima o título especificado na diretiva "$HEAD".

c) $NOLIST

Esta diretiva inibe a emissão das listagens produzidas pelo montador, liberando apenas a listagem de erros.

6.5.4 Diretivas Especiais

As diretivas especiais são comandos que indicam ao montador a preparação de informações que serão utilizadas em tempo de carga e execução:

a) $INOUT

A presença deste comando permite ao usuário invocar as rotinas de entrada e saída padrão, descritas na seção 6.6.2.

b) $MONITOR

Esta diretiva informa ao montador que o programa deve ser monitorado em tempo de execução. As tarefas executadas pelo montador para permitir a monitoração estão descritas na seção 7.3.4.

6.6 Entrada e Saída

A linguagem de montagem não dispõe de comandos especiais para entrada e saída, devendo ser usados os comandos "LOAD" e "STORE" referindo-se aos endereços de memória onde é mapeada a entrada e saída. A área virtual de mapeamento, com seus endereços em hexadecimal, é mostrada

na figura 6.6.

O usuário PCIR deverá armazenar ou acessar a área de dados da memória principal conforme a operação seja de entrada ou saída. O dispositivo de entrada ou saída (disquete, vídeo, impressora, teclado), bem como os parâmetros necessários a cada operação devem ser informados na palavra de endereço 7BAE (hexadecimal) que está reservada para este fim. Esta palavra de parâmetros tem os seguintes campos:

7BAE(15) - Bit de execução. Deve ser 1 para que a operação de entrada e saída seja realizada. Este bit é desligado (tornado zero) automáticamente pela máquina após a execução da operação.

7BAE(14 : 3) - Parâmetros de entrada e saída. Os parâmetros necessários a cada operação estão especificados na seção 6.6.1.

7BAE(2 : 0) - Operação de entrada e saída.As operações de entrada e saída estão especificados na seção 6.6.1.

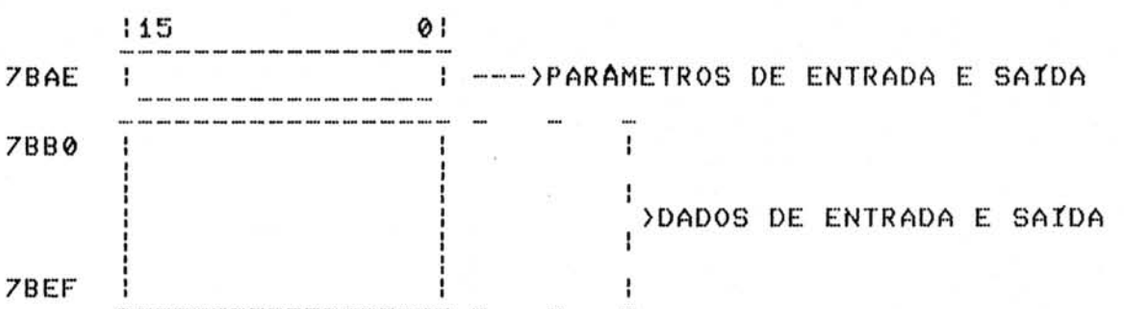


FIGURA 6.6 Área Virtual de Entrada e Saída

6.6.1 Operações de Entrada e Saída

Conforme a operação de entrada ou saída, devem ser fornecidos os parâmetros necessários e a palavra 7BAE tem uma configuração específica a cada operação, conforme ilustrado nas figuras 6.7, 6.8, 6.9, 6.10 e 6.11.

FIGURA 6.7 Leitura em Disquete

FIGURA 6.8 Gravação em Disquete

FIGURA 6.9 Impressão

FIGURA 6.10 Leitura de Teclado

FIGURA 6.11 Exibição no Vídeo

6.6.2 Rotinas de Entrada e Saída Padrão

Com a finalidade de facilitar a comunicação do usuário com os dispositivos de entrada e saída foram construídas rotinas padrão para leitura em disquete, gravação em disquete e impressão de linha.

Estas rotinas estão disponíveis em uma área do disquete conhecida pelo montador.

O usuário deverá informar que estas rotinas serão usadas em seu programa, através da diretiva especial "$INOUT".

Quando encontra esta diretiva, o montador apenas grava informações de carga em relação a estas rotinas, como se fizessem parte do programa do usuário.

Estas rotinas têm nomes especiais, começando com o símbolo "$" e a invocação é feita normalmente pelo

comando de chamada de procedimento "CALL".

Os parâmetros exigidos pelas rotinas conforme especificados na seção 6.6.1, são enviados nas palavras seguintes à instrução de chamada.

A seguir mostra-se através de um esqueleto de programa genérico, a chamada a estas rotinas e a passagem de parâmetros.

```
      CALL  $(nome da rotina)
ROTIO EQU   *
      RES   N          ; N é o número de parâmetros
      ST    =ROTIO , (registrador contendo parâmetro 1)
      ST    =ROTIO + 1, (registrador contendo parâmetro 2)
      ST    =ROTIO + 2, (registrador contendo parâmetro 3)
      .     .      .     .      .      .
      .     .      .     .      .      .
      .     .      .     .      .      .
      ST    =ROTIO + N-1, (registrador contendo parâmetro N)
```

6.7 Tabelas

Um programa montador exige a estruturação de um grande volume de informações em tabelas.

Sabe-se que uma parte considerável do tempo de execução é utilizada na manipulação destas estruturas, as quais constituem um ponto crítico na eficiência do programa.

Observando este aspecto, estas estruturas foram criteriosamente projetadas levando-se em conta a natureza da tabela, as operações mais freqüentes e as instruções disponíveis na linguagem; procurando estabelecer um

compromisso entre a agilização do processo de montagem e a economia do espaço de armazenamento.

As tabelas usadas neste montador podem ser divididas, quanto à natureza, em tabelas estáticas que são criadas independentes do programa fonte a ser traduzido e em tabelas dinâmicas que merecem um tratamento diferenciado em relação às anteriores.

Como tabelas estáticas temos a tabela de códigos de máquina, tabela de diretivas de montagem, diretivas de listagem e diretivas especiais. Além destas tabelas que serão apenas consultadas durante a montagem, existem outras que são criadas dinamicamente pelo montador, quais sejam, tabela de símbolos e tabela de referências cruzadas.

As próximas seções apresentam a organização, estrutura e método de acesso que foram implementados nas tabelas do montador PCIR.

6.7.1 Organização da Tabela de Códigos de Máquina

A tabela de códigos de máquina está organizada sequencialmente, em ordem alfabética pelo mnemônico da instrução.

6.7.1.1 Estrutura

A estrutura da tabela de códigos de máquina e seu conteúdo é mostrada na tabela 6.8.

TABELA 6.8 Códigos de Máquina

MNEMÔNICO	ICODIGO	FORMATO
ADD	0000	0203
ADDC	0100	0203
ADDI	0400	0203
AND	0500	0203
BF	3000	0202
BR	3C00	0101
CALL	5000	0101
DEC	0A00	0101
EXTB	1100	0202
INC	0800	0101
INSB	1000	0203
LD	6000	0202
MOV	3C00	0202
NOP	3C00	0000
OR	0600	0203
POP	7000	0202
PUSH	5200	0203
RET	7000	0000
ROLC	3800	0202
RORC	3A00	0202
SEQ	1A00	0203
SGE	2000	0203
SGT	2200	0203
ST	4000	0200
SUB	0200	0203
SUBC	0300	0203
XOR	0700	0203

6.7.1.2 Definição dos Campos

MNEMÔNICO - campo de duas palavras que contém o mnemônico das instruções de máquina reconhecidas pelo montador.

ICODIGO - campo de uma palavra que contém o código da instrução, representado em hexadecimal. Este código representa os bits correspondentes aos campos "IM", "CODINST" e "OP" da instrução de máquina conforme seção 2.5.

A formação deste código foi feita considerando o campo "IM" igual a zero. Após é feita a justaposição dos três campos mencionados acima, obtendo-se desta forma um código ocupando uma palavra. Faz-se uma inversão dos bytes desta palavra e obtém-se o código mostrado na tabela.

FORMATO - campo de uma palavra representado em hexadecimal que informa o número de operadores da instrução. O primeiro byte contém o número de operadores da instrução sem imediato (instruções de uma palavra) e o segundo byte contém o número de operadores das instruções com imediato (instruções de duas palavras).

6.7.1.3 Método de Acesso

A tabela de códigos de máquina será acessada a cada instrução do programa fonte via pesquisa binária, usando como argumento de pesquisa o mnemônico da instrução.

O método caracteriza-se por reduzir a tabela à metade a cada comparação. Este processo pode ser demonstrado através de uma árvore de decisão binária,

conforme figura 6.12, na qual o valor de um nodo é o mnemônico da instrução e o endereço é o índice da entrada da tabela a ser pesquisada. Um caminho da raiz até qualquer nodo representa a sequência de comparações exigidas para obter um mnemônico ou então determinar a sua ausência.

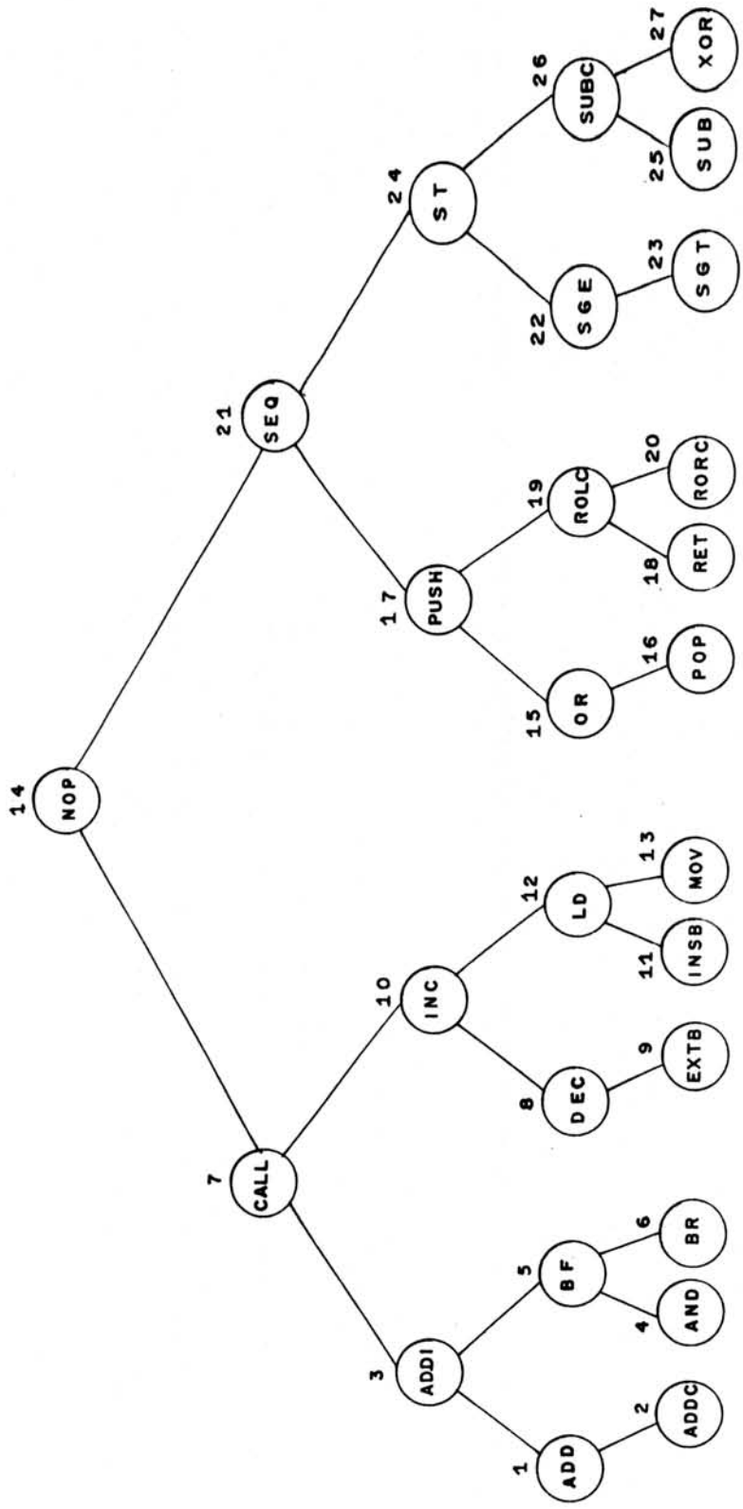


FIGURA 6.12 Árvore de Decisão

6.7.2 Organização da Tabela de Diretivas

As diretivas de montagem, listagem e especiais estão organizadas seqüencialmente em uma única tabela, segundo a probabilidade de ocorrência, ou seja, as mais usadas estão colocadas no início da tabela para minimizar o tempo de pesquisa.

6.7.2.1 Estrutura

A estrutura da tabela de diretivas e seu conteúdo é mostrada na tabela 6.9.

TABELA 6.9 Diretivas

DIRETIVAS	CÓDIGO
RES	F000
DATA	F001
EQU	F002
ORG	F003
END	F004
$HEAD	F005
$PAGE	F006
$NOLIST	F007
$INOUT	F008
$MONITOR	F009

6.7.2.2 Definição dos Campos

DIRETIVA - campo de quatro palavras que especifica o mnemônico da diretiva.

CÓDIGO - campo de uma palavra que contém um código que identifica o tipo de diretiva, representado em hexadecimal.

6.7.2.3 Método de Acesso

A tabela de diretivas será acessada pelo nome da diretiva, quando falhar a busca na tabela de códigos de máquina.

O pequeno número de diretivas e a certeza de que para as diretivas mais utilizadas serão necessários, no pior caso, quatro acessos não permitem que se faça uma gerência mais sofisticada desta tabela. Logo, o método de acesso será seqüencial, garantindo assim a simplicidade da implementação sem prejuízo da eficiência.

6.7.3 Organização da Tabela de Referências Cruzadas

A tabela de referências cruzadas é usada para manter as linhas fontes em que o símbolo foi referido.

Devido à extensão, esta tabela está armazenada em disquete e organizada em forma de uma lista encadeada para cada símbolo, cujo nodo cabeça está presente na tabela de símbolos nos campos "ELOPRIM" e "ELOULT", conforme seção 6.7.4.

6.7.3.1 Estrutura

A estrutura da tabela de referências cruzadas é mostrada na tabela 6.10.

TABELA 6.10 Tabela de Referências Cruzadas

LINHA	ELO

6.7.3.2 Definição dos Campos

LINHA - campo de uma palavra que contém a linha fonte em que o símbolo foi referido.

ELO - campo de uma palavra que contém um ponteiro para a posição da tabela em que o símbolo foi novamente referido.

6.7.3.3 Método de Acesso

A tabela de referências cruzadas é acessada cada vez que um símbolo é referido, através do campo "ELOULT" do nodo cabeça que está armazenado como atributo do símbolo na tabela de símbolos.

A linha de referência é inserida seqüencialmente na próxima posição disponível da tabela.

6.7.4 Organização da Tabela de Símbolos

A tabela de símbolos é usada para manter os nomes simbólicos (símbolos) com os seus atributos.

As operações efetuadas sobre esta tabela são a

inserção e a pesquisa, exigindo que a tabela seja manipulada a cada operação.

Devido às freqüentes pesquisas, a gerência desta tabela é decisiva na eficiência do montador. Então, a tabela está organizada pelo cálculo do endereço do argumento de pesquisa que é o próprio símbolo. Este processo é conhecido como "HASHING" /KNU 73/.

O endereço do símbolo é calculado através de aplicações sucessivas da função "XOR" sobre os caracteres (S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8) não brancos até obter um resultado de uma palavra de 16 bits, sobre a qual é efetuada a adição de seus bytes, conseguindo-se dessa forma o endereço. Este esquema está ilustrado na figura 6.13, onde v_1 e v_2 representam valores intermediários de 16 bits.

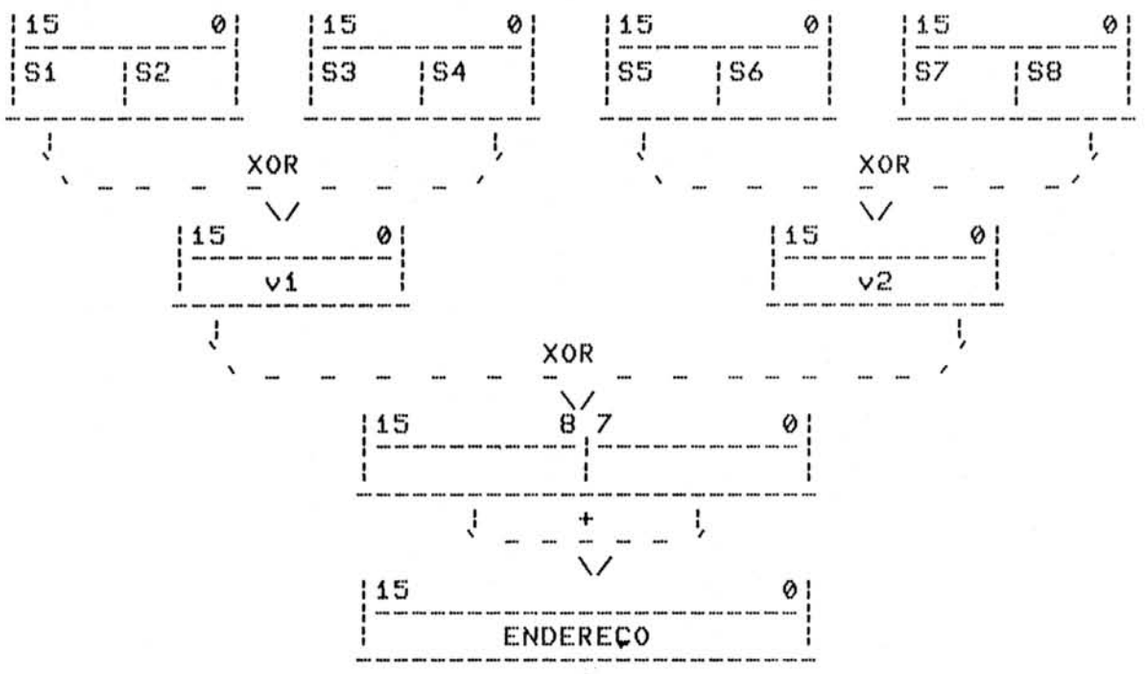


FIGURA 6.13 Esquema do Cálculo do Endereço

6.7.4.1 Estrutura

A tabela de símbolos é composta de duas partes, uma área principal e uma área de overflow onde são armazenados seqüencialmente os símbolos sinônimos que causaram colisão na área principal. Os símbolos sinônimos constituem uma lista encadeada, onde o primeiro nodo da lista aparece na área principal da tabela. A estrutura da tabela de símbolos é mostrada na tabela 6.11.

TABELA 6.11 Tabela de Símbolos

SÍMBOLO	PONTOV	TIPO	LINHA	VALOR	ELOPRIM			ELOULT			
					TP	SP	PP	TU	SU	PU	
											ÁREA PRINCIPAL
											ÁREA DE OVERFLOW

6.7.4.2 Definição dos Campos

SIMBOLO - campo de quatro palavras que contém o nome do símbolo.

PONTOV - campo de 9 bits que contém o ponteiro para o símbolo sinônimo na área de overflow.

TIPO - campo de 2 bits que contém o código que identifica o tipo do símbolo conforme tabela 6.12.

TABELA 6.12 Tipos de Símbolos

TIPO	CÓDIGO
definido absoluto	00
definido relativo	10
indefinido	10

LINHA - campo de uma palavra que contém a linha fonte onde o símbolo foi definido.

VALOR - campo de uma palavra que contém o valor do símbolo.

ELOPRIM - campo de uma palavra que contém o ponteiro para a posição da tabela de referências cruzadas que contém a primeira linha onde o símbolo foi referido. Este campo está dividido em três subcampos que contém a trilha, o setor e a posição dentro do setor da tabela de referências cruzadas.

ELOULT - campo de uma palavra que contém o ponteiro para a posição da tabela de referências cruzadas que contém a última linha onde o símbolo foi referido. Este campo está dividido em três subcampos que contém a trilha, o setor e a posição dentro do setor da tabela de referências cruzadas.

6.7.4.3 Método de Acesso

A tabela de símbolos é acessada a cada requisição de inserção ou recuperação de atributos usando o método de acesso via hashing que caracteriza-se pela utilização do esquema ilustrado na figura 6.13.

Este método transforma o símbolo em um número utilizável como endereço para a tabela. Se o símbolo não é encontrado na área principal, a pesquisa segue pelos ponteiros de encadeamento até que a busca tenha sucesso ou até concluir-se que o símbolo não está contido na tabela. O acesso é ilustrado na figura 6.14.

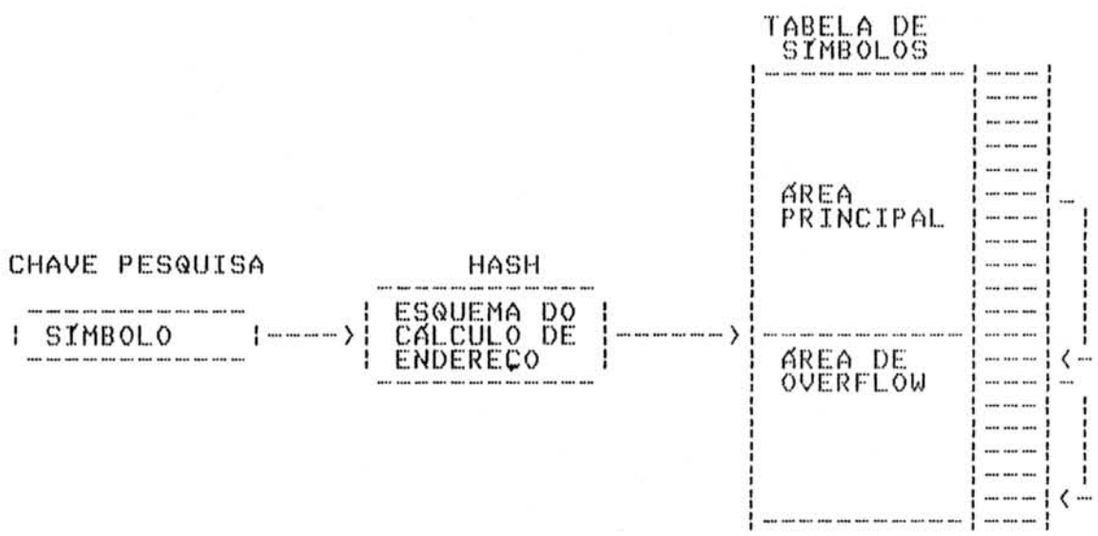


FIGURA 6.14 Acesso via Hashing

6.8 Código Fonte versus Código Intermediário

O programa fonte que será submetido ao montador está armazenado em disquete, onde cada linha fonte de 80 caracteres ocupa 40 palavras de um setor. As 24 palavras restantes de cada setor são utilizadas pelo montador para gravar o código intermediário, ao lado do código fonte. A configuração do código intermediário é mostrada na figura 6.15.

FIGURA 6.15 Código Intermediário

6.8.1 Definição dos Campos

ENDER - campo de uma palavra que contém o endereço da instrução.

CODINST - campo de uma palavra que contém o código da instrução, conforme especificado nas seções 6.7.1.1, 6.7.2.1 e 6.7.4.1.

ERRO - campo de uma palavra que acusa através de seus bits, a ocorrência e o tipo do erro na linha fonte, conforme especificado na tabela 6.13.

TABELA 6.13 Tabela de Erros

PRIMEIRO BYTE	SIGNIFICADO
1xxx xxxx	INDICA QUE EXISTE ERRO
x1xx xxxx	FORMATO DA INSTRUÇÃO INVÁLIDO
xx1x xxxx	RÓTULO INVÁLIDO
xxx1 xxxx	CÓDIGO DE OPERAÇÃO INVÁLIDO
xxxx 1xxx	OPERANDO 1 INVÁLIDO
xxxx x1xx	OPERANDO 2 INVÁLIDO
xxxx xx1x	OPERANDO 3 INVÁLIDO
xxxx xxx1	SÍMBOLO INDEFINIDO
SEGUNDO BYTE	**SIGNIFICADO**
1xxx xxxx	SÍMBOLO REDEFINIDO
x1xx xxxx	OPERANDO DEVE SER ABSOLUTO
xx1x xxxx	OPERANDO DEVE SER RELATIVO
xxx1 xxxx	CONSTANTE INVÁLIDA
xxxx 1xxx	EXPRESSÃO INVÁLIDA
xxxx x1xx	DELIMITADOR INVÁLIDO
xxxx xx1x	NOME SIMBÓLICO INVÁLIDO
xxxx xxx1	CADEIA DE CARACTERES INVÁLIDA

OPERANDOS - campo de 22 palavras dividido em vários subcampos, conforme o conteúdo da linha fonte. A linha fonte pode conter:

a) Instrução de Máquina ou Diretivas de montagem (ORG, RES, EQU) - Neste caso, o campo "OPERANDOS" é dividido em 6 subcampos de uma palavra cada um, agrupados dois a dois formando o par "TIPO OPERANDO - VALOR" que tem o conteúdo especificado na tabela 6.14, onde o tipo está representado em hexadecimal. A figura 6.16 mostra a configuração do campo "OPERANDOS" do código intermediário para estas instruções.

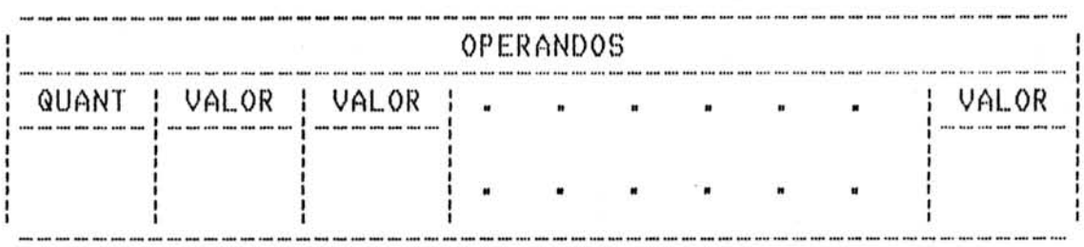
TABELA 6.14 Conteúdo do Campo "OPERANDOS"

TIPO OPERANDO	SIGNIFICADO
0000	ausência de operando
4001	valor de registrador
4002	ponteiro para TS (registrador)
4003	ponteiro para CF (registrador)
8001	valor imediato
8002	ponteiro para TS (imediato)
8003	ponteiro para CF (imediato)
2000	constante especial

OPERANDOS						
TIPO	VALOR	TIPO	VALOR	TIPO	VALOR	SEM SIGNIFICADO

FIGURA 6.16 Código Intermediário do Campo "OPERANDOS"

b) Diretiva de Montagem ("DATA") - Neste caso, o campo "OPERANDOS" é dividido em um campo de uma palavra, denominado "QUANT" e tantas quantas forem as constantes especificadas na linha fonte até um máximo de 21, definem um campo "VALOR" de uma palavra, conforme figura 6.17.



QUANT - quantidade de constantes

VALOR - valor da constante

FIGURA 6.17 Código Intermediário do Campo "OPERANDOS" da Diretiva "DATA"

c) Outras - Todas as outras instruções definem o campo "OPERANDOS" sem significado.

6.9 Listagens

As listagens emitidas normalmente pelo montador são mostradas utilizando-se um trecho de programa como exemplo. A figura 6.18 apresenta o código fonte com o respectivo código objeto representado em hexadecimal. Se há erro na linha fonte, a palavra de erros descrita na seção 6.8, representada aqui em hexadecimal aparece logo abaixo da linha que causou o erro, onde um asterisco substitui o número da linha.

Na figura 6.19 são especificadas as mensagens de erro correspondentes às configurações possíveis da palavra de erros.

A figura 6.20 contém a tabela de símbolos em ordem alfabética, complementada pela tabela de referências cruzadas, com os atributos do símbolo e as linhas onde o símbolo foi referido.

LINHA	CÓDIGO FONTE		ENDEREÇO	CÓDIGO OBJETO
1		ORG 10	0000	----
2	R1	EQU 1	000A	----
3	R2	EQU 2	000A	----
4	R3	EQU 3	000A	----
5	PC	EQU 15	000A	----
6	SP	EQU 14	000A	----
7	ST	EQU 13	000A	----
8	LOOP	MOV SP,=PILHA	000A	BCE0
				0029
9		MOV ST,=0	000C	BCD0
				0000
10		CALL INISUB	000E	CAEF
				0022
11		NOP	0010	3C00
12		MOVE 4,R1	0011	3C00
* ERRO 9000				
13		MOV SP,=H'0044	0012	BCE0
				0044
14		MOV R1,=H'00A	0014	BC10
				000A
15		PUSH SP,R1	0016	4AE1
16		MOV R2,=H'00B	0017	BC20
				000B
17		PUSH SP,R2	0019	4AE2
18		POP 5,SP	001A	585E
19		POP 5,SP	001B	585E
20		SEQ R1,R2	001C	1A12
21		BF R1,SAIDA	001D	B0F1
				002C
22		BR LOOP	001F	BCF0
				000A

continua...

LINHA	CÓDIGO FONTE		ENDEREÇO	CÓDIGO OBJETO
23		NOP	0021	3C00
24	INISUB	MOV R1,=C'AB	0022	BC10
				4142
25		MOV R2,=C'XY	0024	BC20
				5859
26		INSB R1,R2	0026	1012
27		RET	0027	58FE
28		NOP	0028	3C00
29	PILHA	RES 3	0029	----
30	SAIDA	NOP	002C	3C00
31		END	----	----

FIGURA 6.18 Listagem do Programa Fonte e Programa Objeto

SIGNIFICADO DOS ERROS		
PRIMEIRO BYTE	HEXA	SIGNIFICADO
1xxx xxxx	80	EXISTE ERRO
x1xx xxxx	40	FORMATO DA INSTRUÇÃO INVÁLIDO
xx1x xxxx	20	RÓTULO INVÁLIDO
xxx1 xxxx	10	CÓDIGO DA OPERAÇÃO INVÁLIDO
xxxx 1xxx	08	OPERANDO 1 INVÁLIDO
xxxx x1xx	04	OPERANDO 2 INVÁLIDO
xxxx xx1x	02	OPERANDO 3 INVÁLIDO
xxxx xxx1	01	SÍMBOLO INDEFINIDO
SEGUNDO BYTE	HEXA	SIGNIFICADO
1xxx xxxx	80	SÍMBOLO REDEFINIDO
x1xx xxxx	40	OPERANDO DEVE SER ABSOLUTO
xx1x xxxx	20	OPERANDO DEVE SER RELATIVO
xxx1 xxxx	10	CONSTANTE INVÁLIDA
xxxx 1xxx	08	EXPRESSÃO INVÁLIDA
xxxx x1xx	04	DELIMITADOR INVÁLIDO
xxxx xx1x	02	SÍMBOLO INVÁLIDO
xxxx xxx1	01	CADEIA DE CARACTERES INVÁLIDA

FIGURA 6.19 Mensagens de Erros

SÍMBOLO	VALOR	LINHA DE DEFINIÇÃO	LINHAS DE REFERÊNCIA
INISUB	0022	24	10
LOOP	000A	8	22
PC	000F	5	
PILHA	0029	29	8
R1	0001	2	12, 14, 15, 20, 21, 24, 26
R2	0002	3	17, 20, 25, 26
R3	0003	4	
SAIDA	002C	30	21
SP	000E	6	8, 13, 15, 17, 18, 19
ST	000D	7	9

FIGURA 6.20 Listagem de Símbolos e Referências Cruzadas

7 IMPLEMENTAÇÃO DO MONTADOR PCIR

Este capítulo descreve os procedimentos de implementação do montador PCIR, baseados em /BAR 72/ e /CAL 79/.

7.1 Introdução

O montador PCIR está escrito na linguagem de montagem descrita no capítulo 6.

Devido à pouca disponibilidade de memória, o montador está dividido em três módulos:

- Primeira passagem: É o módulo responsável pela análise sintática do programa fonte, pela construção das tabelas de símbolos e de referências cruzadas e pela gravação do código intermediário.

- Segunda passagem: É o módulo responsável pela geração do código objeto, utilizando as estruturas criadas na primeira passagem.

- Listador: É o módulo responsável pela emissão das listagens. O listador acessa o código fonte, o código objeto, a tabela de símbolos e a tabela de referências cruzadas para emitir as listagens descritas na seção 6.9.

A figura 7.1 apresenta o fluxo geral do montador PCIR.

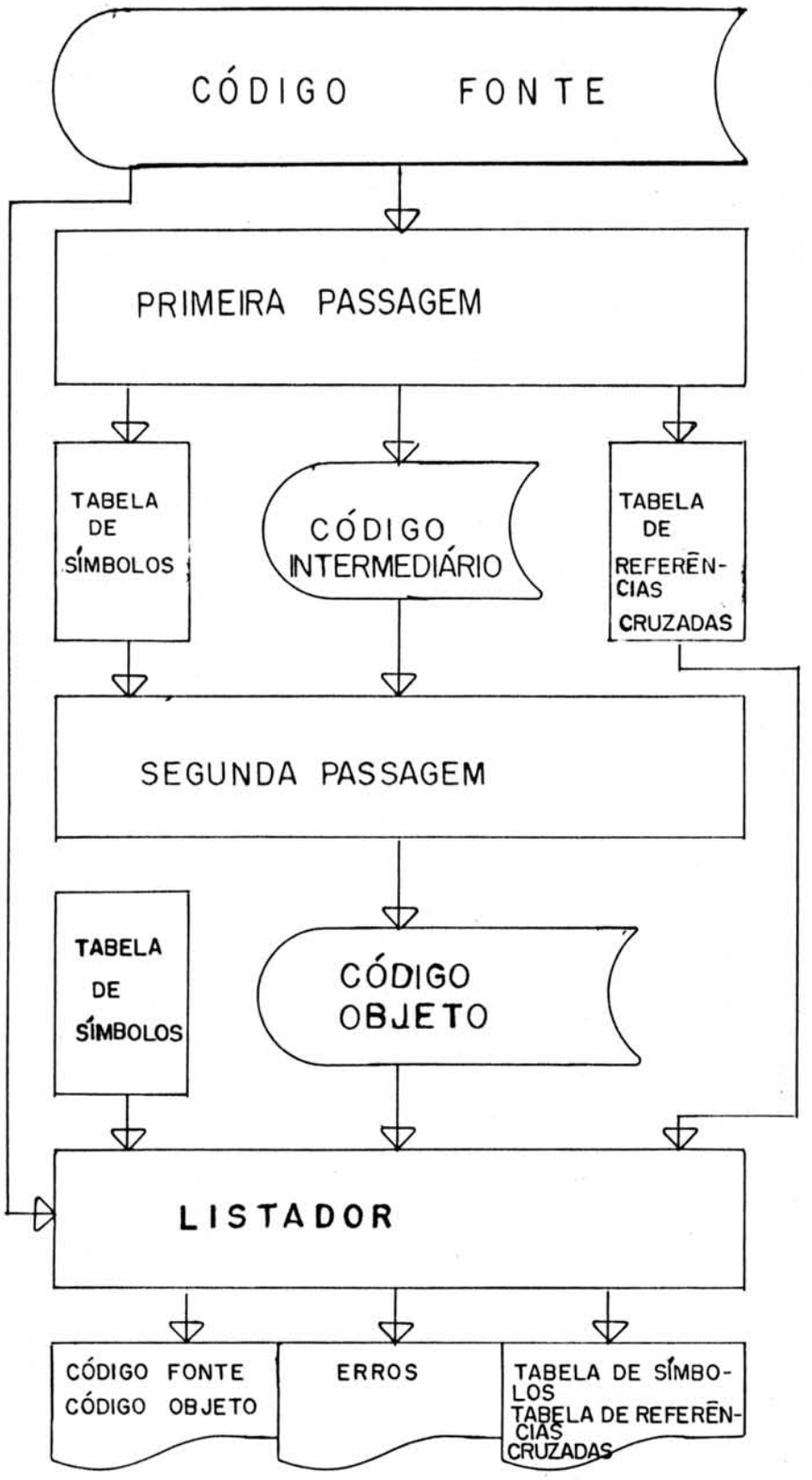


FIGURA 7.1 Fluxo Geral do Montador PCIR

7.2 Primeira Passagem

A primeira passagem tem como funções principais a construção da tabela de símbolos e a gravação do código intermediário. São também tarefas deste módulo a análise do programa fonte, a verificação dos erros e a formação da tabela de referências cruzadas.

A escolha das funções pertinentes a este módulo foi baseada na premissa de executar as tarefas o mais cedo possível sem sacrificar a velocidade e procurando dar condições para um tratamento uniforme do código intermediário pelo módulo que realiza a segunda passagem.

7.2.1 Análise da Linha Fonte

A linha fonte é trazida para a memória e é percorrida caractere a caractere formando o que chamamos de elemento sintático da linguagem. São considerados elementos sintáticos, o mnemônico de uma instrução, o rótulo e os operandos nas suas diversas representações (constante, expressão, nome simbólico, cadeia de caracteres, referência a contador de posição).

Cada elemento é extraído da linha fonte e armazenado em um array de caracteres para ser posteriormente analisado conforme será descrito nas seções 7.2.2, 7.2.3 e 7.2.4.

A cada um dos elementos sintáticos corresponde um diagrama de estados e uma matriz de transição de estados. Para uma análise completa de todos os caracteres da linha fonte são utilizados alguns diagramas auxiliares que não

têm correspondência com os elemento sintáticos.

Nos diagramas, cada estado é representado por um nó e as possíveis transições por linhas direcionadas e rotuladas pelos caracteres que provocam estas transições. Os nomes dos diagramas são compostos por três letras, começando com a letra "D". Todos os diagramas começam no estado inicial "I".

As matrizes de transição de estados de cada diagrama apresentam nas colunas a classe do caractere e nas linhas todos os estados possíveis daquele diagrama, sendo que "R" significa o reconhecimento do elemento sintático e "E" significa erro no elemento em processo de reconhecimento.

As seguintes convenções foram adotadas quanto à classe do caractere:

C - significa letra C
H - significa letra H
L - significa qualquer letra
LH - significa qualquer letra de "A" a "F"
D - significa qualquer dígito
0 - significa dígito "0"
1 - significa dígito "1"
FL - significa fim da linha fonte
≠ - significa qualquer caractere diferente dos especificados
Ø - significa branco

7.2.1.1 Diagramas de Estados dos Elementos Sintáticos

Os diagramas de estados de cada um dos elementos sintáticos da linha fonte com as respectivas matrizes de transição de estados estão apresentados conforme relação abaixo:

a) Diagrama de reconhecimento de rótulo (DRR), nas figuras 7.2 e 7.3.

b) Diagrama de reconhecimento de código da instrução (DRI), nas figuras 7.4 e 7.5.

c) Diagrama de reconhecimento de operandos (DRO), dividido em partes, nas figuras 7.6, 7.7, 7.8, 7.9, 7.10, 7.11, 7.12, 7.13, 7.14, 7.15 e 7.16.

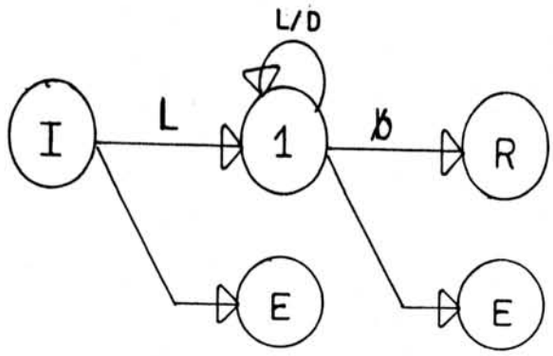


FIGURA 7.2 DRR

	L	D	Ø	≠
I	1	E	E	E
1	1	1	R	E

FIGURA 7.3 Matriz de Transição de Estados do DRR

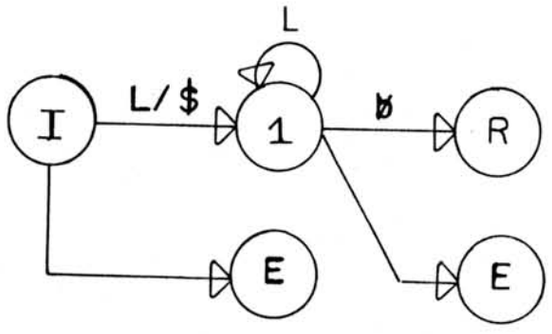


FIGURA 7.4 DRI

	L	Ø	$	≠
I	1	E	1	E
1	1	R	E	E

FIGURA 7.5 Matriz de Transição de Estados do DRI

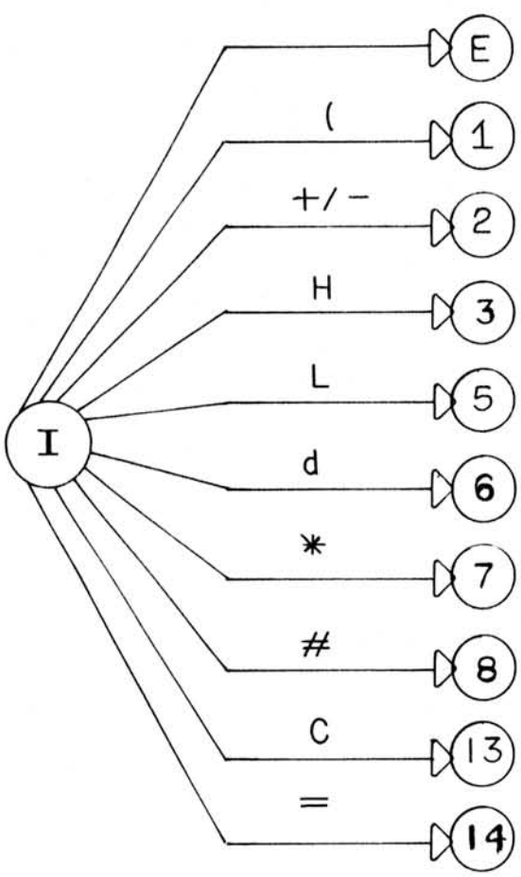


FIGURA 7.6 DRO a Partir do Estado Inicial

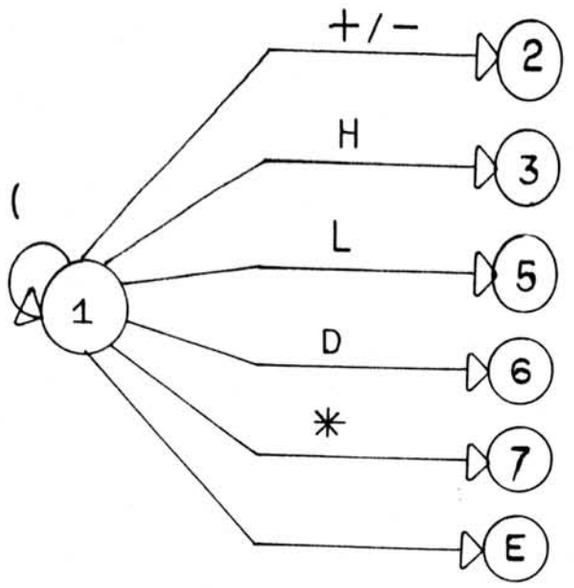


FIGURA 7.7 DRO a Partir do Estado 1

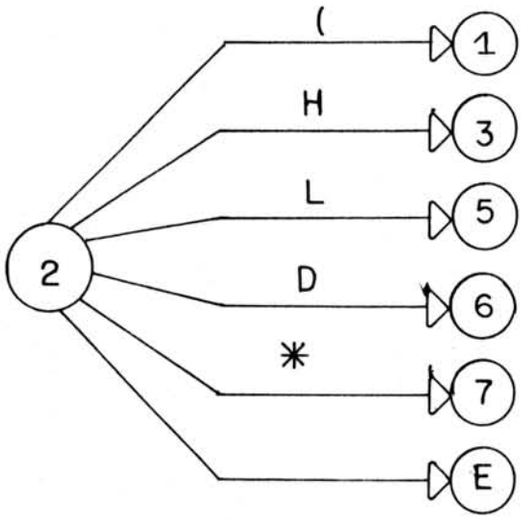


FIGURA 7.8 DRO a Partir do Estado 2

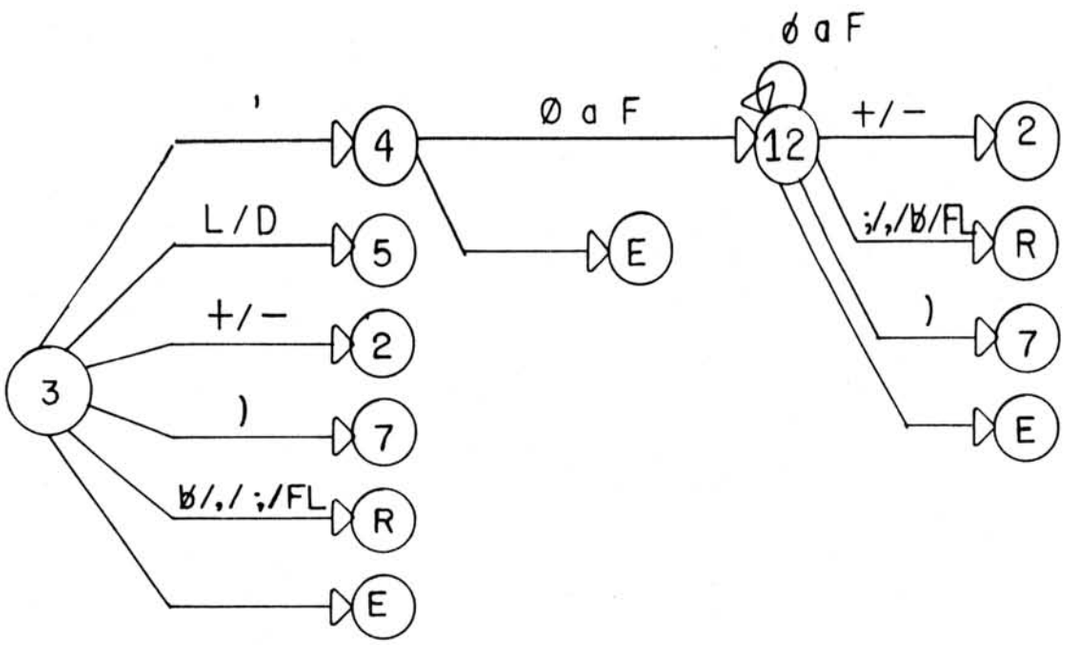


FIGURA 7.9 DRO a Partir do Estado 3

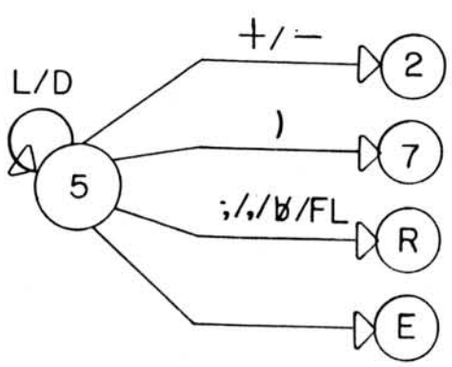


FIGURA 7.10 DRO a Partir do Estado 5

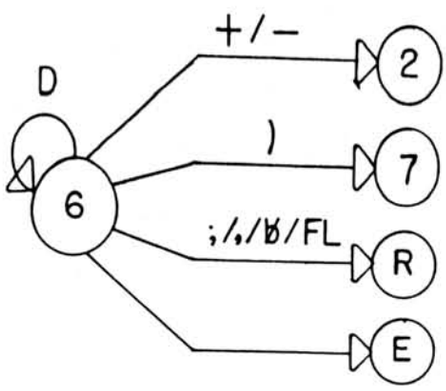


FIGURA 7.11 DRO a Partir do Estado 6

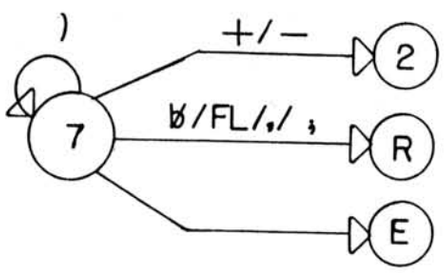


FIGURA 7.12 DRO à Partir do Estado7

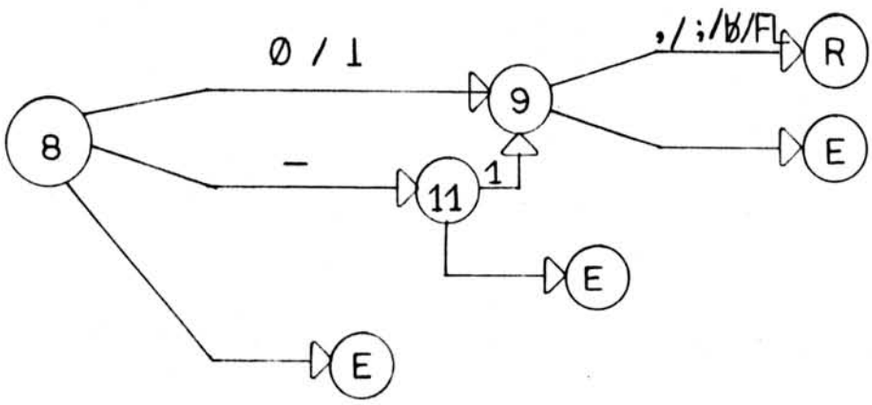


FIGURA 7.13 DRO à Partir do Estado 8

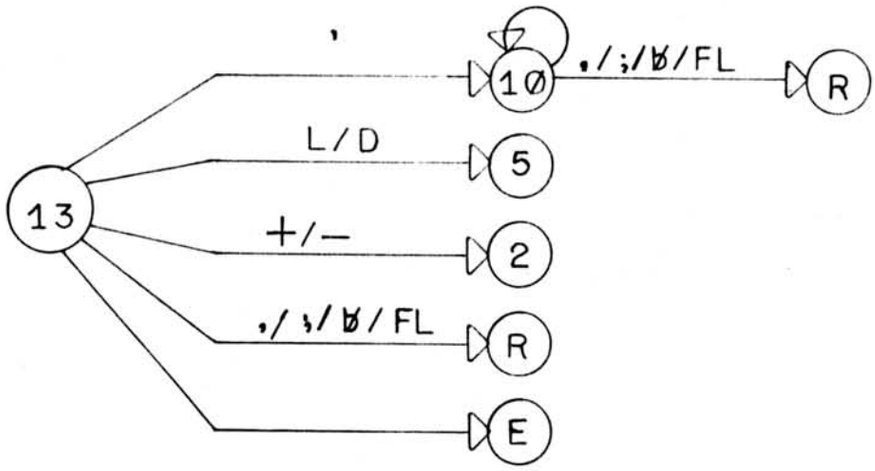


FIGURA 7.14 DRO a Partir do Estado 13

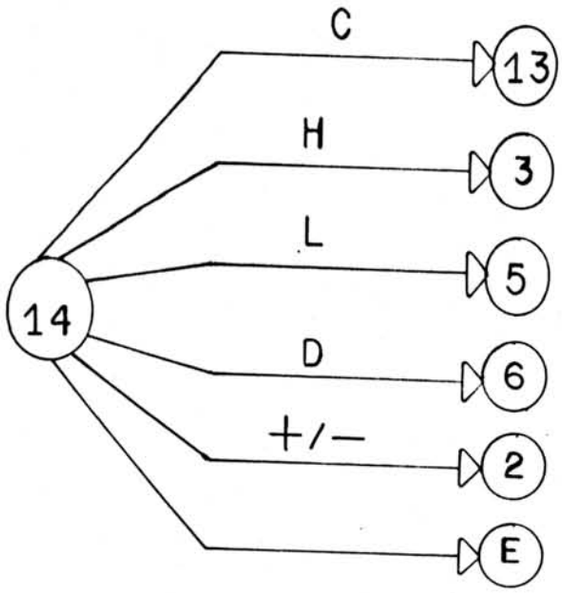


FIGURA 7.15 DRO a Partir do Estado 14

	L	D	*	#	C	=	H	+	−	(	)	,	;	`	LH	Ø	1	Ø	FL	≠
I	5	6	7	8	13	14	3	2	2	1	E	E	E	E	E	E	E	E	E	E
1	5	6	7	E	E	E	3	2	2	1	E	E	E	E	E	E	E	E	E	E
2	5	6	7	E	E	E	3	E	E	1	E	E	E	E	E	E	E	E	E	E
3	5	5	E	E	E	E	E	2	2	E	7	R	R	4	E	E	E	R	R	E
4	E	12	E	E	E	E	E	E	E	E	E	E	E	E	12	12	12	E	E	E
5	5	5	E	E	E	E	E	2	2	E	7	R	R	E	E	E	E	R	R	E
6	E	6	E	E	E	E	E	2	2	7	E	R	R	E	E	E	E	R	R	E
7	E	E	E	E	E	E	E	2	2	E	7	R	R	E	E	E	E	R	R	E
8	E	E	E	E	E	E	E	E	11	E	E	E	E	E	9	9	E	E	E	E
9	E	E	E	E	E	E	E	E	E	E	E	R	R	E	E	E	E	R	R	E
10	10	10	10	10	10	10	10	10	10	10	10	R	R	10	10	10	10	R	R	10
11	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	9	E	E	E
12	E	12	E	E	E	E	2	2	E	E	7	R	R	E	12	12	12	R	R	E
13	5	5	E	E	E	E	E	2	2	E	E	R	R	10	E	E	E	R	R	E
14	5	6	E	E	13	E	3	2	2	E	E	E	E	E	E	E	E	E	E	E

FIGURA 7.16 Matriz de Transição de Estados do DRO

7.2.1.2 Diagramas de Estados (Auxiliares)

Os diagramas de estados que auxiliam na análise da linha fonte, com as respectivas matrizes de transições de estados estão apresentados conforme relação abaixo:

a) Diagrama de reconhecimento de branco (DRB), nas figuras 7.17 e 7.18.

b) Diagrama de reconhecimento de comentário (DRC), nas figuras 7.19 e 7.20.

c) Diagrama de reconhecimento de literal (DRL), nas figuras 7.21 e 7.22.

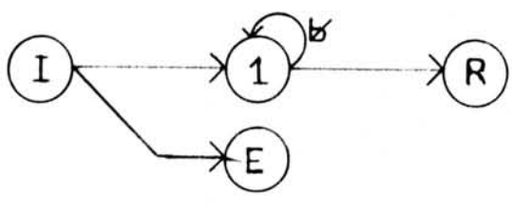


FIGURA 7.17 DRB

	ø	≠
I	1	E
1	1	R

FIGURA 7.18 Matriz de Transição de Estados do DRB

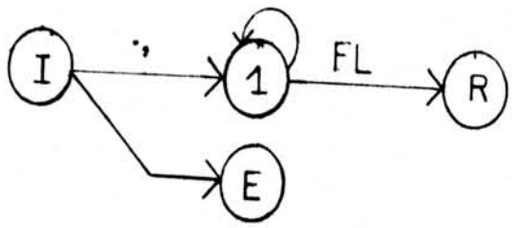


FIGURA 7.19 DRC

	;	FL	≠
I	1	E	E
1	1	R	1

FIGURA 7.20 Matriz de Transição de Estados do DRC

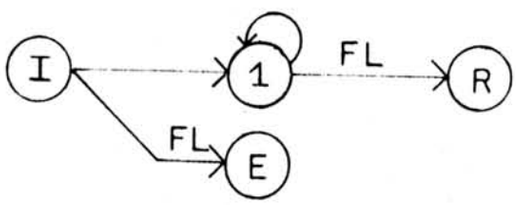


FIGURA 7.21 DRL

	FL	≠
I	E	1
1	R	1

FIGURA 7.22 Matriz de Transição de Estados do DRL

7.2.1.3 Algoritmo da Análise da Linha Fonte

```
* CHAR - caractere da linha fonte
inicio
  obtem CHAR;
  se CHAR =`Ø´  então inicio
                        DRB;
                        caso CHAR =
                          `;´  : DRC;
                          `$´  : inicio
                                   DRI;
                                   DRT;
                                 fim;
                          outro: inicio
                                   DRI;
                                   POSDRI;
                                 fim;
                        fim;
    senão caso CHAR =
            `;´  : DRT;
            `$´  : DRT;
            outro: inicio
                     DRR;
                     se CHAR = `Ø´ então DRB    senão ERRO;
                     DRI;
                     POSDRI;
                   fim;
          fim;
  se not RLF  então inicio
                     se CHAR = `Ø´ então DRB;
                     caso CHAR =
                       `;´  : DRC;
                       FL   : nop;
                       outro: ERRO;
                     fim;
                   fim;
fim
```

7.2.1.4 Algoritmos das Rotinas Utilizadas na Análise da Linha Fonte

```
POSDRO (analisa a linha após o reconhecimento de operando)
inicio
 RLF <-- false;
 caso CHAR =
   `Ø': inicio
          DRB;
          caso CHAR =
            FL   : RLF <-- true;
            `;' : inicio
                    DRC;
                    RLF <-- true;
                  fim;
            `,' : inicio
                    obtem CHAR;
                    se CHAR = `Ø' então DRB;
                    DRO ; RLF <--false;
                  fim;
            outro: inicio
                    ERRO (delimitador inválido);
                    RLF <-- false;
                  fim;
          fim;
    `,': inicio
           obtem CHAR;
           se CHAR = `Ø' então DRB;
           DRO;
           RLF <-- false;
         fim;
    `;': inicio
           DRC;
           RLF <-- true;
         fim;
 fim;
fim
```

```
POSDRI (analisa a linha após o reconhecimento da instrução)
inicio
  caso CHAR =
     ";"  : DRC;
     "b"  : inicio
              DRB;
              caso CHAR =
                ";"  : DRC;
                FL   : nop;
                outro: inicio
                         DRO;
                         POSDRO;
                         se not RLF então POSDRO;
                       fim;
              fim;
     outro: inicio
              ERRO;
              DRO;
              POSDRO;
              se not RLF então POSDRO;
            fim;
  fim;
fim
```

7.2.2 Análise do Código da Instrução

O código da instrução, ou seja, o mnemônico que representa a operação é extraido da linha fonte e é utilizado como argumento de pesquisa para acessar primeiramente a tabela de instruções de máquina através de uma pesquisa binária conforme descrito na seção 6.7.1.3.

Se o mnemônico for encontrado, o código correspondente é gravado no campo "CODINST" do código intermediário. Caso contrário, haverá então uma nova pesquisa, agora seqüencial, na tabela de diretivas. se encontrado nesta tabela, o código correspondente é então gravado no campo "CODINST" do código intermediário.

Se a busca a estas duas tabelas não obtiver sucesso então a operação é considerada inválida e o bit 12 da palavra de erros é levado ao estado lógico 1 para acusar a invalidade da operação.

A partir da análise da operação, a linha fonte é classificada como linha de instrução de máquina ou linha de diretiva. Estes dois tipos de linha fonte terão seus conteúdos analisados e avaliados de forma diferente, produzindo um código intermediário que permitirá um tratamento uniforme, na segunda passagem.

7.2.3 Análise das Instruções de Máquina

A linha fonte que contém uma instrução de máquina terá cada elemento analisado de acordo com a sua função, conforme figura 7.23.

RÓTULO	OPERAÇÃO	OPERANDOS

FIGURA 7.23 Elementos da Instrução de Máquina

7.2.3.1 Análise do Rótulo

O rótulo é extraido da linha fonte e constitui o argumento de pesquisa para acessar a tabela de símbolos pelo método do cálculo do endereço abordado na seção 6.7.4.

O valor do símbolo que representa o rótulo é computado através do contador de posições (endereço) atual.

A pesquisa tem três resultados possíveis que passaremos a descrever em paralelo com a ação a ser tomada.

Se o símbolo não se encontra na tabela, é inserido juntamente com o valor. E o tipo fica estabelecido como definido relativo.

Se o símbolo já se encontra na tabela e o tipo é indefinido então apenas o valor é inserido e o tipo é modificado para definido relativo.

Se o símbolo já se encontra na tabela mas o tipo é definido então é considerado símbolo redefinido e o bit 7 da palavra de erros é levado ao estado lógico 1, acusando a redefinição.

7.2.3.2 Análise dos Operandos

O operando é extraído da linha fonte e é classificado conforme o tipo de dado pelo qual está representado. A avaliação é realizada pelo avaliador do tipo de dado apropriado, conforme será descrito na seção 7.2.3.3.

Há um tratamento de exceção quanto ao segundo operando das instruções aritméticas e lógicas. Se este operando for uma constante especial, embora possa ser tratado como os demais, exige um tratamento adicional sobre o código da instrução obtido da tabela de códigos de máquina já que a instrução deve ser transformada na instrução de máquina "OPK" apresentada na seção 2.5.3, a qual é transparente ao programador. Desta forma, deve ser somado o valor "0800" (hexadecimal) ao código que está gravado no campo "CODINST" do código intermediário.

Outro procedimento acrescentado à análise dos operandos é a verificação de um valor imediato utilizado como operando, já que a presença do imediato determina que a instrução não é de formato básico, exigindo a modificação do código da instrução gravado no campo "CODINST" do código intermediário, ao qual deve ser somado o valor "8000" (hexadecimal).

7.2.3.3 Avaliação dos Tipos de Dados

Os tipos de dados que podem ser utilizados na representação dos operandos das instruções de máquina são avaliados através dos procedimentos descritos nas próximas seções.

7.2.3.3.1 Avaliação de Nome Simbólico

O nome simbólico ou símbolo usado na representação dos operandos é utilizado como argumento de pesquisa para a tabela de símbolos.

A pesquisa pode resultar em situações diferentes que passamos a descrever em paralelo com a ação a ser tomada.

Se o símbolo já se encontra na tabela e está classificado como definido então o seu valor é recuperado e gravado no campo de operando apropriado do código intermediário com a indicação de que o operando é um valor, associada com o tipo do operando que pode ser registrador ou imediato.

Se o símbolo já se encontra na tabela mas está classificado como indefinido então o ponteiro (endereço da tabela) da localização do símbolo é gravado no campo de operando apropriado do código intermediário, juntamente com a indicação de que o operando é um ponteiro para a tabela de símbolos associada com o tipo do operando.

Se o símbolo não se encontra na tabela então seu endereço é calculado conforme seção 6.7.4 e é inserido como indefinido. A gravação no código intermediário prossegue como no caso anterior.

7.2.3.3.2 Avaliação de Expressões

As expressões usadas na representação dos operandos das instruções de máquina não são resolvidas na primeira passagem. No código intermediário, no campo de operando apropriado é gravado o ponteiro (posição da linha fonte onde inicia a expressão) e a indicação de que o operando é um ponteiro para a linha fonte, associado ao tipo registrador ou imediato.

7.2.3.3.3 Avaliação de Constantes

As constantes, na base decimal, hexadecimal ou do tipo caractere usadas na representação dos operandos são transformadas em um valor de 16 bits. Este valor é gravado no campo de operando apropriado do código intermediário, com a indicação de que o operando é um valor associada ao tipo de operando que pode ser registrador ou imediato.

7.2.4 Análise das Diretivas

As próximas seções descrevem a análise do rótulo e dos operandos das diretivas que necessitam um tratamento diferenciado em relação às instruções de máquina. Quanto ao campo de operação já foi efetuada a análise conforme descrito na seção 7.2.2.

A linha fonte que contém uma diretiva terá seus elementos analisados de acordo com a função, conforme figura 7.24.

RÓTULO	DIRETIVA	OPERANDOS

FIGURA 7.24 Elementos da Diretiva

7.2.4.1 EQU

O rótulo da diretiva constitui o argumento de pesquisa para acessar a tabela de símbolos pelo método do cálculo do endereço abordado na seção 6.7.4.3.

O valor do símbolo utilizado como rótulo é computado através do operando conforme será descrito mais adiante.

Se o símbolo não se encontra na tabela, é inserido juntamente com o valor e o tipo definido absoluto ou relativo, dependendo do operando.

Se o símbolo já se encontra na tabela e o tipo é indefinido então seu valor é inserido e o tipo é modificado para definido relativo ou absoluto, dependendo do tipo do operando.

Se o símbolo já se encontra na tabela mas o tipo é definido, então é considerado "símbolo redefinido" e os bits 7 e 11 da palavra de erros são levados ao estado lógico 1, acusando a redefinição do símbolo e a invalidade do operando.

Esta diretiva pode ter como operando uma constante decimal ou hexadecimal, uma expressão, um nome simbólico ou uma referência ao contador de posição (endereço atual). O operando utilizado deve permitir a avaliação imediata.

Se o operando for uma constante esta é transformada em um valor de 16 bits, o qual é atribuído ao símbolo utilizado como rótulo.

Se o operando for uma expressão, esta deve conter somente símbolos já definidos. Desta forma a expressão usada como operando desta diretiva pode excepcionalmente ser resolvida na primeira passagem. O procedimento de resolução encontra-se descrito na seção 7.3.1.2. O resultado obtido é atribuído ao símbolo utilizado como

rótulo. Se a expressão contém algum símbolo indefinido, não é realizada a resolução e os bits 8, 3 e 11 da palavra de erros são levados ao estado lógico 1, acusando o símbolo indefinido e a invalidade da expressão e do operando.

7.2.4.2 RES

O símbolo utilizado como rótulo nesta diretiva é tratado da mesma forma que o rótulo de qualquer instrução de máquina, conforme descrito na seção 7.2.3.1.

Esta diretiva permite como operando uma constante decimal ou hexadecimal, uma expressão ou um nome simbólico. O operando deve permitir a avaliação imediata e é analisado da mesma forma que o operando da diretiva "EQU", na seção 7.2.4.1, com a diferença que o valor obtido da avaliação é somado ao contador de posição, incrementando assim o endereço atual.

7.2.4.3 DATA

O símbolo utilizado como rótulo nesta diretiva é tratado da mesma forma que o rótulo de qualquer instrução de máquina, conforme descrito na seção 7.2.3.1.

Os operandos desta diretiva são constantes especificadas em qualquer base aceita pelo montador ou uma constante especificada por uma cadeia de caracteres.

Estas constantes são transformadas cada uma em um valor de 16 bits gravado nos campos "VALCONST" do código intermediário.

As constantes representadas pelo tipo de dado cadeia de caracteres recebem um tratamento especial, ou seja, a cada dois caracteres é computado um valor de 16 bits, até um máximo de 8 caracteres. Se o número de caracteres especificados é ímpar, então é inserido um caractere branco no final.

No campo "NUMCONST" do código intermediário é gravada a quantidade de palavras ocupadas pelas constantes especificadas na linha fonte. Esta quantidade é somada ao contador de posição, incrementando o endereço atual.

7.2.4.4 ORG

Esta diretiva admite como operando, uma constante (decimal ou hexadecimal) ou um nome simbólico.

Estes operandos são avaliados conforme seção 7.2.3.3.

7.2.4.5 Diretivas de Listagem

A única diretiva de listagem que necessita tratamento quanto ao operando é a diretiva "$HEAD" que tem o título especificado. Este título é transformado em representação interna dos caracteres que o compõe, sendo gravado no campo "OPERANDOS" do código intermediário.

7.2.4.6 Diretivas Especiais

Apenas a indicação da presença das diretivas especiais é gravada no código intermediário, quando da análise do código da instrução.

A especificação da diretiva "$INOUT" deixa o montador "avisado" para aceitar as chamadas das rotinas padrão de entrada e saída descritas na seção 6.6.2.

7.2.5 Geração da Tabela de Símbolos e de Referências Cruzadas

A tabela de símbolos e a tabela de referências cruzadas são geradas junto com os procedimentos já descritos nas seções anteriores. A cada linha fonte é verificada a presença de nomes simbólicos representando o rótulo ou o operando da instrução.

Este símbolo constitui o argumento de pesquisa para a tabela de símbolos a qual é acessada aplicando-se o método do cálculo do endereço descrito na seção 6.7.4.3 e que é ilustrado na figura 7.26 com um exemplo real, a partir dos símbolos encontrados no trecho do programa fonte que aparece na figura 7.25.

Para formar a tabela de referências cruzadas, a tabela de símbolos é acessada a fim de se obter o nodo cabeça da lista encadeada de referências ao símbolo. O número da linha fonte onde o símbolo aparece como rótulo é colocado no campo "LINHA" da tabela de símbolos. E o número da linha fonte em que o símbolo é referido é inserido na próxima posição disponível da tabela de referências

cruzadas. Esta posição é mantida durante toda a primeira passagem, no campo "ELOTRC" o qual é atualizado a cada nova inserção.

O campo "ELOULT" da tabela de símbolos é analisado. Se o seu conteúdo é nulo significa que a lista está vazia e o conteúdo do campo "ELOTRC" é atribuído aos campos "ELOPRIM" e "ELOULT" da tabela de símbolos. Se o conteúdo do campo "ELOULT" não é nulo então seu valor é utilizado para acessar a tabela de referências cruzadas. E o conteúdo do campo "ELOTRC" é atribuído ao campo "ELO" da tabela de referências cruzadas e ao campo "ELOULT" da tabela de símbolos.

A figura 7.27 ilustra o processo mostrando o conteúdo dos campos que são afetados pelas operações descritas, partindo do trecho de programa fonte que aparece na figura 7.25.

LINHA FONTE	CÓDIGO FONTE		
.			
.			
.			
118		ORG	H`400
119	R4	EQU	4
120	INIC	ADD	1,2
121		SUB	R4,#1
122	XXXX	SEQ	R4,2
123		BF	4,INIC
124	CALCENDT	MOV	R4,=3
125		MOV	5,=4
126		MOV	6,=5
127	R5	EQU	5
128	R6	EQU	6
129		SEQ	R5,R4
130		BF	5,XXXX
131		BR	INIC
132	SAIDA	ADD	R7,R8
.			
.			
.			

FIGURA 7.25 Trecho de um Programa Fonte Sendo Avaliado

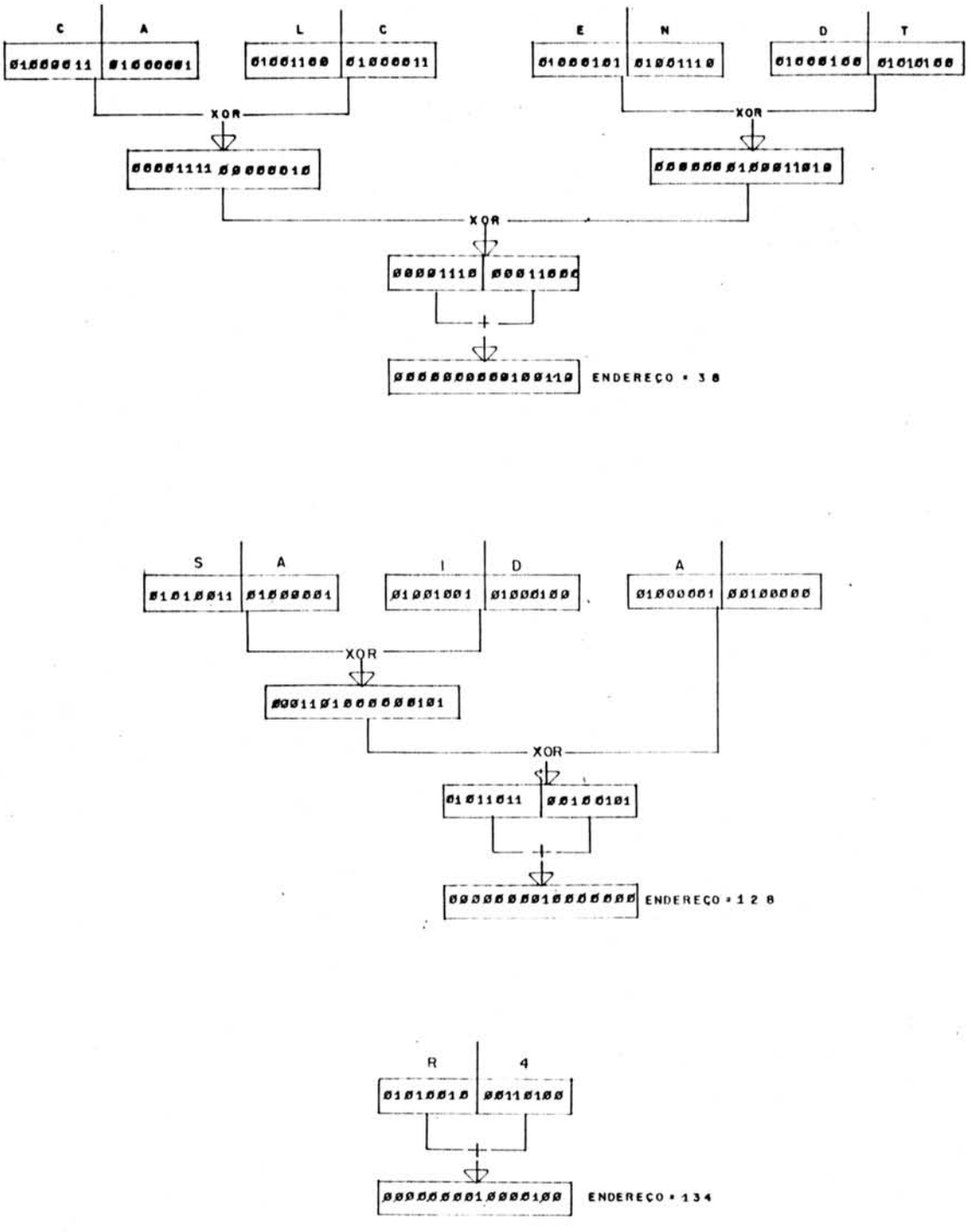


FIGURA 7.26 Cálculo de Endereço dos Símbolos "CALCENDT", "SAIDA" e "R4"

TABELA DE SÍMBOLOS

ENDEREÇO	SÍMBOLO	PONTOV	TIPO	LINHA	VALOR	ELOPRIM	ELOULT
0	XXXX	--	01	122	1026	7	7
13	INIC	--	01	120	1024	3	8
38	CALCENDT	--	01	124	1029	0	0
128	SAIDA	--	01	132	1040	0	0
134	R4	--	00	119	4	1	6
135	R5	--	00	127	5	5	5
136	R6	--	00	128	6	0	0
137	R7	--	10	132	--	9	9
138	R8	--	10	132	--	10	10

TABELA DE REFERÊNCIAS CRUZADAS

	LINHA	ELO
1	121	2
2	122	4
3	123	8
4	124	6
5	129	0
6	129	0
7	130	0
8	131	0
9	132	0
10	132	0
11		

ELOTRC
11

FIGURA 7.27 Formação da Tabela de Símbolos e de Referências Cruzadas

7.2.6 Geração do Código Intermediário

A geração do código intermediário que é a atividade principal da primeira passagem foi detalhada nas seções anteriores em relação a cada campo da linha fonte e do tratamento das particularidades inerentes a cada situação.

Adicionalmente, a figura 7.28 mostra o código intermediário gerado para o trecho de programa fonte já utilizado na seção anterior para a formação da tabela de símbolos e de referências cruzadas.

O algoritmo da geração do código intermediário para os operandos é apresentado na seção 7.2.6.1.

Para dar uma visão geral dos procedimentos realizados neste estágio é apresentado na seção 7.2.6.2 o algoritmo da primeira passagem no qual os detalhes foram propositadamente omitidos em benefício da clareza.

LINHA FONTE		CÓDIGO INTERMEDIÁRIO							
		ENDER	CODINST	TIPO	VALOR	TIPO	VALOR	TIPO	VALOR
	ORG H`400	1024	F003	5	1024	0			
R4	EQU 4	1024	F002						
INIC	ADD 1,2	1024	0000	1	1	1	2	0	
	SUB R4,#1	1025	0A00	1	4	4	0	0	
XXXX	SEQ R4,2	1026	1A00	1	4	1	2	0	
	BF 4,INIC	1027	B000	1	13	1	4	5	1024
CALCENDT	MOV R4,=3	1029	BC00	1	4	5	3	0	
	MOV 5,=4	1031	BC00	1	5	5	4	0	
	MOV 6,=5	1033	BC00	1	6	5	5	0	
R5	EQU 5	1035	F002						
R6	EQU 6	1035	F002						
	SEQ R5,R4	1035	1A00	1	5	1	4	0	
	BF 5,XXXX	1036	B000	1	13	1	5	5	1026
	BR INIC	1038	BC00	1	13	5	1024	0	
SAIDA	ADD R7,R8	1040	0000	2	137	2	138	0	

FIGURA 7.28 Código Intermediário Gerado

7.2.6.1 Algoritmo de Geração do Código Intermediário para os Operandos

```
* RG - registrador
* IM - imediato
* CE - constante especial
* SD - símbolo definido
* SI - símbolo indefinido
* EX - expressão
* PONTFONTE - ponteiro de caracteres do código fonte
inicio
  se existe OPERANDO
     então inicio
             caso OPERANDO =                  * TIPO OPERANDO *
               RG: CI.TIPO <-- H'4000;
               IM: CI.TIPO <-- H'8000;
               CE: CI.TIPO <-- H'2000;
             fim;
             caso OPERANDO =                   * TIPO DE DADO *
               SD: inicio
                     CI.TIPO <-- CI.TIPO + 1;
                     CI.VALOR <-- TS.VALOR
                   fim;
               SI: inicio
                     CI.TIPO <-- CI.TIPO + 2;
                     TS.VALOR <-- HASH (SIMBOLO)
                   fim;
               EX: inicio
                     CI.TIPO <-- CI.TIPO + 3;
                     CI.VALOR <-- PONTFONTE (EX)
                   fim;
           fim;
         fim
     senão CI.TIPO <-- 0;
fim
```

7.2.7 Algoritmo da Primeira Passagem

```
* CONTLINHA - contador de linha fonte
* CONTPOS - contador de posições (endereço)
* NUMCONST - contador de constantes na linha fonte
inicio
  CONTPOS <-- 0;
  CONTLINHA <-- 0;
  repetir
    obtem LINHA FONTE;
    CONTLINHA <-- CONTLINHA + 1;
    extrai RÓTULO;
    extrai OPERAÇÃO;
    extrai OPERANDOS;
    pesquisa TM;
    se achou
       então inicio
               se existe RÓTULO
                  então inicio
                          VALOR RÓTULO <-- CONTPOS;
                          insere RÓTULO na TS
                        fim;
               avalia OPERANDOS;
               gera CÓDIGO INTERMEDIÁRIO;
               CONTPOS <-- CONTPOS + TM.FORMATO;
             fim
```

```
        senão inicio
                pesquisa TD;
                se achou então  caso DIRETIVA =
                                    EQU   : TRATA-EQU;
                                    ORG   : TRATA-ORG;
                                    DATA  : TRATA-DATA;
                                    RES   : TRATA-RES;
                                    $HEAD: TRATA-$HEAD;
                                    outra: ;
                                fim
                    senão ERRO (instrução inválida);
             fim;
  até OPERAÇÃO = "END";
  SEGUNDA PASSAGEM;
fim


TRATA-EQU
inicio
  avalia OPERANDO;
  VALOR RÓTULO <-- VALOR OPERANDO;
  pesquisa TS;
  se achou então TS.VALOR <-- VALOR RÓTULO
           senão insere RÓTULO na TS;
fim


TRATA-ORG
inicio
  se existe RÓTULO então   insere RÓTULO na TS;
  avalia OPERANDO;
  gera CÓDIGO INTERMEDIÁRIO;
  CONTPOS <-- VALOR OPERANDO;
fim
```

```
TRATA-DATA
inicio
  NUMCONST <-- 0;
  se existe RÓTULO
     então insere RÓTULO na TS;
  enquanto houver CONSTANTE faça
       inicio
         avalia CONSTANTE;
         gera CÓDIGO INTERMEDIÁRIO;
         NUMCONST <-- NUMCONST + 1
       fim;
  CONTPOS <-- CONTPOS + NUMCONST;
fim


TRATA-RES
inicio
  se existe RÓTULO
     então insere RÓTULO na TS;
  avalia OPERANDO;
  gera CÓDIGO INTERMEDIÁRIO;
  CONTPOS <--CONTPOS + VALOR OPERANDO;
fim


TRATA-$HEAD
inicio
  avalia LITERAL;
  gera CÓDIGO INTERMEDIÁRIO;
fim
```

7.3 Segunda Passagem

Este módulo tem como função principal a geração do código objeto a partir da avaliação do código intermediário.

A avaliação tem inicio com a obtenção do campo "CODINST" do código intermediário. Este campo contém o tipo da linha intermediária, direcionando o seu tratamento adequado.

A linha intermediária pode conter uma instrução de máquina, uma diretiva de montagem, uma diretiva de listagem ou uma diretiva especial.

As próximas seções descrevem a avaliação conforme o tipo da linha intermediária.

7.3.1 Avaliação das Instruções de Máquina

Estas instruções têm seu código objeto parcialmente formado no campo "CODINST", contendo o código da instrução e o código da operação.

Os operandos são tratados conforme o campo "TIPO" constituído por 6 bits que têm o significado mostrado na figura 7.29.

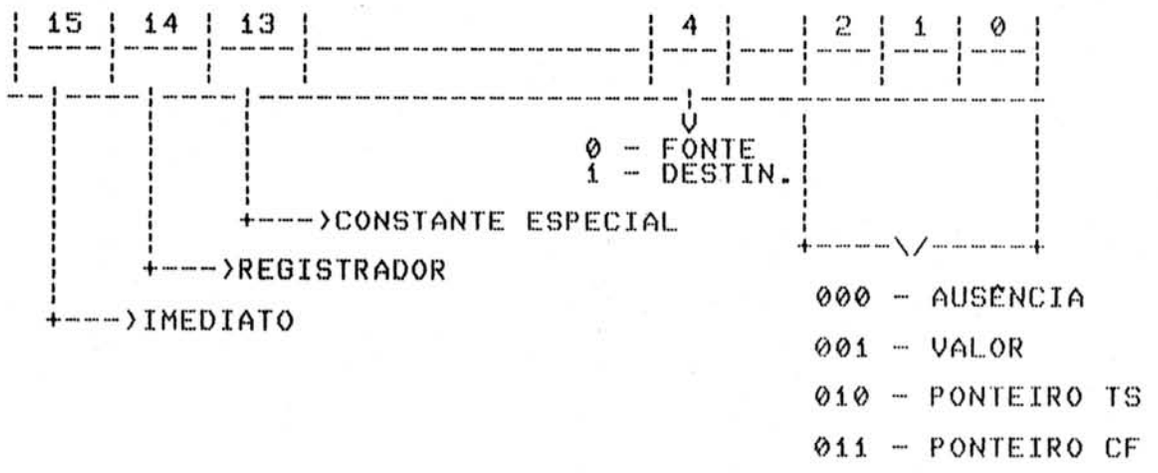


FIGURA 7.29 Tipos dos Operandos

7.3.1.1 Avaliação dos Operandos

Os operandos existentes no código intermediário são avaliados para obter o valor correspondente e são avaliados quanto ao tipo de operando que pode ser registrador, imediato ou constante especial.

A avaliação quanto ao valor do operando é feita verificando se o conteúdo do campo "VALOR" no código intermediário é o próprio valor do operando, se é um ponteiro para a tabela de símbolos ou se é um ponteiro para o código fonte.

Se o código intermediário contém um ponteiro para a tabela de símbolos, esta é acessada e o valor do símbolo é recuperado. Se o símbolo permanece indefinido na tabela então o bit correspondente ao operando e o bit 8 da palavra de erros são levados ao estado lógico 1 indicando a invalidade do operando.

Se o código intermediário contém um ponteiro para

o código fonte, isso significa que o operando foi representado através de uma expressão a qual é resolvida conforme seção 7.3.1.2 para obter o valor.

Se o código intermediário contém o próprio valor do operando, não é necessário nenhum procedimento especial.

O valor obtido desta avaliação é colocado nos campos que correspondem ao registrador destino, registrador fonte ou imediato conforme a posição do operando e o tipo da instrução.

Se o operando foi classificado como registrador, então o valor obtido deve ser um número entre 0 e 15. Do contrário o bit correspondente ao operando na palavra de erros é levado ao estado lógico 1 indicando a invalidade do operando.

Se o operando foi classificado como constante especial, a consistência já foi realizada na primeira passagem.

Além disso os operandos são consistidos quanto ao tipo absoluto ou relativo, de acordo com a instrução. Caso detectada a inadequação do tipo ao operando, os bits correspondentes na palavra de erros são levados ao estado lógico 1 indicando a invalidade do operando.

7.3.1.2 Resolução das Expressões

As expressões são avaliadas em um único exame da esquerda para a direita, com o uso de duas pilhas. Uma pilha de operadores (OPPILHA) que mantém os operadores

aritméticos mais (+) e menos (-) e o abre parêntese "(". Uma pilha de operandos (VALPILHA) que mantém os valores dos símbolos, das constantes e resultados intermediários.

Durante o exame, cada vez que uma constante ou símbolo é encontrado, seu valor é empilhado em VALPILHA. Cada vez que um "abre parêntese" é encontrado, é empilhado na OPPILHA.

Quando um operador aritmético é encontrado, é empilhado na OPPILHA, porém antes de empilhar o operador, o elemento do topo da OPPILHA é examinado. Se este for um operador é então retirado e a operação que ele representa é realizada entre os dois valores do topo da VALPILHA. Estes dois valores são retirados da pilha e substituídos pelo resultado da operação.

Quando um "fecha parêntese" é encontrado, o elemento do topo da OPPILHA é retirado e a operação que ele representa é realizada.

Prossegue este processo de retirada e realização das operações representadas pelos operadores da OPPILHA até que um "abre parêntese" é encontrado, sendo então o par de parênteses eliminado.

A ordem de realização das operações é posicional (da esquerda para direita) e com prioridade sobre os parênteses mais internos.

O exame e a realização das operações prossegue desta forma até encontrar um branco, um ponto e vírgula ou o fim da linha fonte. Quando um destes delimitadores é encontrado, o procedimento é o mesmo de quando encontrado

um operador e já existe um operador no topo de OPPILHA.

Inicialmente, antes de começar o exame da expressão, a palavra de erros é consultada para obter a validade ou não da expressão . Se a expressão está inválida não é feita a resolução mas se está sintaticamente correta então as pilhas são iniciadas vazias e a resolução é realizada conforme foi descrito, obtendo-se um valor que deverá ser verificado quanto à consistência com o tipo de operando que representa. Esta avaliação é competência do avaliador de operandos que invocou a rotina de resolução de expressões.

7.3.2 Avaliação das Diretivas de Montagem

A diretiva de montagem RES provoca a reserva de palavras de memória. O valor do operando desta diretiva determina a quantidade de palavras reservadas que ficam sem especificação.

A diretiva de montagem DATA também provoca a reserva de palavras de memória porém com os valores especificados pelos campos "VALOR" do código intermediário. O campo "QUANT" do código intermediário determina a quantidade de palavras reservadas com os valores especificados.

A diretiva ORG, indica através do valor do seu operando, o endereço de carga do código objeto gerado a partir da sua especificação.

A cada ORG encontrado é iniciado um novo módulo interno (segmento de programa com endereço de carga). As

informações relativas à carga são gravadas no setor 1 do disquete, nos campos apropriados, conforme especificado no capítulo 5.

A diretiva END faz com que o montador complemente as informações de carga e execução gravadas no setor 1.

7.3.3 Avaliação das Diretivas de Listagem

É criada na segunda passagem uma área de dados globais de nome "GLIST" para manter informações relativas às listagens. Esta área será posteriormente consultada pelo módulo que emite as listagens.

A diretiva $PAGE faz com que o número da linha fonte seja colocado no campo apropriado da área "GLIST".

A diretiva $HEAD faz com que o título seja armazenado no campo "TITULO" da área "GLIST".

A diretiva $NOLIST faz com que o bit de listagem da área "GLIST" seja levado ao estado lógico 1 para inibir a listagem da tabela de símbolos e de referências cruzadas.

7.3.4 Avaliação das Diretivas Especiais

A diretiva $MONITOR indica que a opção de monitoração deve ser ligada quando da execução do programa. Isto é feito colocando o bit de monitoração (BITMON) do setor 1 no estado lógico 1. Esta informação é verificada pelo gerente de módulos objeto quando acionar o emulador.

A diretiva $INOUT faz com que no final da segunda passagem, após encontrar a diretiva END, o montador grave no setor 1 informações relativas à carga das rotinas de entrada e saída padrão, como se estas fossem parte do programa do usuário.

7.3.5 Geração do Código Objeto

O código objeto já está parcialmente formado no campo "CODINST" do código intermediário para cada instrução.

Após a avaliação dos operandos, os valores correspondentes aos campos registrador fonte e registrador destino são inseridos nesta palavra. Se a instrução contém um valor imediato este é armazenado na próxima palavra gerando então um código objeto de duas palavras para a instrução.

O código objeto, à medida que está sendo gerado é gravado no disquete após o programa fonte. E as informações de trilha e setor de localização são gravadas no setor 1 para permitir a carga do programa pelo carregador quando acionado pelo gerente de módulos objeto.

7.3.6 Algoritmo da Segunda Passagem

```
* LI - linha intermediária
* LF - linha fonte
* SI - símbolo indefinido
*  I - índice que acessa o par "TIPO-VALOR" do campo
       OPERANDOS do código intermediário, conforme figura
       7.29
*  VALEXP - valor obtido quando da resolução de uma
            expressão
* VALOB - campo auxiliar que recebe o valor do operando
* RDOB - registrador destino no código objeto CO <7:4>
* RFOB - registrador fonte no código objeto CO <3:0>
* IMOB - valor imediato no código objeto CO <15:0>
* GLIST - área de informações para listagem
* CGIO - indica a presença da diretiva $INOUT
* IM - indica instrução de duas palavras


inicio
  enquanto houver LI faça
    ler LI;
    se LI é instrução de máquina
       então inicio
               para I = 1 até 3 faça
                 inicio
                   BUSCAVAL (I, VALOB);
                   caso CI.OPERANDOS[I] <15:12>
                     0: ;
                     2: RFOB <-- VALOB;
```

```
                    4: inicio
                         se VALOB < 0 ou VALOB > 15
                          então
                            ERRO (operando I inválido)
                          senão
                            se CI.OPERANDOS.TIPO[I](4) = 0
                                 então RFOB <-- VALOB
                                 senão RDOB <-- VALOB;
                       fim;
                    8: inicio
                         IM <-- true;
                         IMOB <-- VALOB;
                       fim;
                  fim;
                fim;
              grava objeto;
              se IM então grava IMOB;
            fim
      senão TRATA-DIRETIVA;
  fim;
fim.


TRATA-DIRETIVA
inicio
  caso diretiva =
    RES: inicio
           BUSCAVAL (1, VALOB);
           para X = 1 até VALOB faça
             inicio
               IMOB <-- 0;
               grava IMOB;
             fim;
         fim;
```

```
    DATA: para X = 1 até CI.QUANT faça
            inicio
              BUSCAVAL (X, VALOB);
              IMOB <-- VALOB;
              grava IMOB;
            fim;

    ORG: grava informações de carga no setor 1;

    END: inicio
           complementa informações de carga;
           liga bit de execução;
           se CGIO
              então grava informações de carga para rotinas
                    de entrada e saída;
         fim;

    $PAGE: GLIST.LINHA <-- número da linha fonte;

    $HEAD: GLIST.TITULO <-- CI.OPERANDOS;

    $NOLIST: GLIST.LIST(15) <-- 1;

    $MONITOR: BITMON <-- 1;

    $INOUT: CGIO <-- true;

  fim;
fim.
```

```
BUSCAVAL (I, VALOB)
inicio
  caso CI.OPERANDOS.TIPO[I](1:0) =
     0: ;

     1: VALOB (-- CI.OPERANDOS.VALOR[I]

     2: inicio
          pesquisa TS;
          se SI então ERRO (operando I inválido)
               senão VALOB (-- TS.VALOR;
        fim;

     3: inicio
          resolve expressão;
          se expressão inválida
             então ERRO (operando I inválido)
             senão VALOB (-- VALEXP;
        fim;
  fim;
fim.
```

7.4 Listador

O terceiro módulo que compõe o montador é o listador, responsável pela emissão das listagens mostradas na seção 6.9.

Este módulo lê de forma balanceada os dois programas (fonte e objeto) e imprime as listagens correspondentes.

Durante a emissão das listagens, o listador obtém informações na área de dados globais "GLIST" para realizar

a impressão conforme especificado pelo usuário.

Este módulo é ainda responsável pela organização alfabética da tabela de símbolos, antes da impressão.

8 MANUAL DO USUÁRIO

Este capítulo descreve os procedimentos necessários para montar e executar um programa PCIR.

8.1 Codificação de Um Programa PCIR

Os programas escritos na linguagem de montagem PCIR, devem seguir as especificações relativas a cada instrução, descritas no capítulo 6.

A seguir apresenta-se algumas regras gerais que devem ser seguidas na codificação das instruções e na especificação das diretivas de listagem e especiais.

Cada instrução deve ser codificada em uma linha de 80 posições e cada linha deve conter obrigatoriamente o mnemônico da instrução, o qual pode iniciar em qualquer posição, exceto na primeira.

Se for utilizado rótulo, este deve ser composto de no máximo 8 caracteres, sendo o primeiro caractere obrigatoriamente alfabético. O rótulo deve iniciar na primeira posição da linha.

Entre o rótulo e o mnemônico da instrução deve haver pelo menos um espaço em branco.

Se a instrução exigir operandos, estes devem ser separados por vírgula e deve haver pelo menos um espaço em branco entre o mnemônico da instrução e o primeiro operando.

A diretiva $HEAD, especifica o título que deve ser impresso a cada troca normal de página ou a cada troca de página provocada pela diretiva $PAGE. Então, se estas duas diretivas forem usadas, a diretiva $HEAD deve preceder a diretiva $PAGE, do contrário nenhum título será impresso na nova página.

A diretiva $NOLIST pode ser especificada em qualquer lugar do programa.

A diretiva $INOUT deve preceder a chamada às rotinas padrão de entrada e saída, do contrário as chamadas serão consideradas inválidas.

A diretiva $MONITOR pode ser especificada em qualquer lugar do programa.

Os comentários podem ser escritos na mesma linha da instrução, antecedidos por um ponto e vírgula ou em uma linha separada onde na primeira posição deve ser colocado o ponto e vírgula, indicando que trata-se de uma linha de comentário e não de uma linha de instrução.

O significado de cada uma das instruções e os tipos de dados permitidos na representação dos operandos estão descritos no capítulo 6.

8.2 Montar um Programa PCIR

O programa PCIR a ser montado deve estar contido em um disquete cujo primeiro setor é reservado para ser gravado pelo montador com informações relativas à carga do programa objeto. Os passos necessários para montar um

programa PCIR são os seguintes:

a) Colocar na unidade 21 o disquete de rótulo "EMUOBJ" que contém o emulador (onde estão embutidos o gerente e o carregador) e acionar a chave IMPL do painel de controle para carregar o emulador. Após carregado, aparece na tela a mensagem "MAQUINA PCIR" e logo após a mensagem "CARGA?".

b) Colocar na unidade 21 o disquete de rótulo "MONOBJ" que contém os 3 módulos do montador e acionar uma tecla para que o primeiro módulo do montador seja carregado.

Após a carga deste módulo (primeira passagem), aparece na tela a mensagem "CARPT" e logo após a mensagem "EX?".

c) Colocar na unidade 21 o disquete contendo o programa fonte a ser montado e acionar uma tecla para que seja realizada a primeira passagem no programa fonte.

Após a execução da primeira passagem aparece na tela a mensagem "EXPT" e logo após a mensagem "CARGA?".

d) Colocar na unidade 21 o disquete de rótulo "MONOBJ" e acionar uma tecla para que o segundo módulo do montador seja carregado.

Após a carga deste módulo (segunda passagem) aparece na tela a mensagem "CARPT" e logo após a mensagem "EX?".

e) Colocar na unidade 21 o disquete contendo o programa fonte que está sendo montado e acionar uma tecla

para que seja realizada a segunda passagem no programa fonte.

Após a execução da segunda passagem, aparece na tela a mensagem "EXPT" e logo após a mensagem "CARGA?".

f) Colocar na unidade 21 o disquete de rótulo "MONOBJ" e acionar uma tecla para que o terceiro módulo seja carregado.

Após a carga deste módulo (listador) aparece na tela a mensagem "CARPT" e logo após a mensagem "EX?".

g) Colocar na unidade 21 o disquete contendo o programa fonte que foi montado e acionar uma tecla para que sejam emitidas as listagens produzidas pelo módulo listador do montador.

8.3 Executar um Programa PCIR

O programa PCIR a ser executado pelo emulador deve estar contido em um disquete cujo primeiro setor contém informações de carga e monitoração que foram gravadas pelo montador.

Os passos necessários para executar um programa PCIR são os seguintes:

a) Colocar na unidade 21 o disquete de rótulo "EMUOBJ" que contém o emulador (onde estão embutidos o gerente e o carregador) e acionar a chave IMPL do painel de controle para carregar o emulador.

Após carregado aparece na tela a mensagem "MAQUINA PCIR" e logo após a mensagem "CARGA?".

b) Colocar na unidade 21 o disquete que contém o programa objeto a ser executado e acionar uma tecla para que o programa seja carregado.

Após a carga do programa aparece na tela a mensagem "CARPT" e logo após a mensagem "EX?".

c) Acionar uma tecla para que o programa seja executado.

Após a execução aparece na tela a mensagem "EXPT".

Obs: Se o programa a ser executado necessita acessar dados que estão gravados em outro disquete, a troca deve ser efetuada antes de acionar uma tecla, já que o acionamento de qualquer tecla significa a permissão para iniciar a execução.

9 CONCLUSÕES

O trabalho mostrou uma experiência de construção paralela de software e hardware atendendo aos objetivos pretendidos.

O propósito inicial de viabilizar o uso da máquina PCIR ainda em fase de projeto foi alcançado através da emulação do microprocessador PCIR em firmware.

O emulador além de comportar-se conforme as especificações de projeto do PCIR foi enriquecido com um monitor que realiza a contabilização das instruções executadas, constituindo-se em uma ferramenta de apoio à tarefa de avaliação do projeto físico.

Destaca-se ainda que foi embutido no emulador o gerente de módulo objeto para a máquina PCIR, destinado à carga e execução dos programas, tornando o emulador independente do sistema de suporte para desenvolvimento de firmware.

O segundo objetivo que era a necessidade de um tradutor foi cumprido através da definição de uma linguagem de montagem e da construção de um montador escrito na própria linguagem de montagem PCIR.

O montador, dessa forma procurou obedecer ao compromisso com a portabilidade já que um dos aspectos importantes desse projeto é a possibilidade de migração do software implementado na máquina emulada para a máquina real.

Durante a fase de estudo para definir um tradutor para a máquina PCIR, desejava-se uma linguagem que apresentasse construções de alto nível para que fosse explorada a potencialidade do conjunto de instruções do PCIR, já que um dos objetivos visados em seu projeto é o de ser uma máquina de suporte a uma máquina virtual de mais alto nivel.

Considerando a complexidade da implementação de um compilador nesta fase de projeto, optou-se por uma linguagem de montagem que possa servir de linguagem alvo de um compilador de uma "Linguagem de Alto Nível Orientada ao PCIR" /ALV 85/, cujas características foram baseadas em /POE 74/ e /TOR 82/.

Embora seja necessário muito trabalho de desenvolvimento de software para tornar a máquina PCIR atraente ao usuário, os elementos já desenvolvidos constituem a base de suporte à continuidade do "ambiente PCIR", nesta Universidade.

ANEXO 1

RESUMO DA LINGUAGEM DE MICROMONTAGEM

- Instruções de desvio:

Estas instruções modificam o valor do MPC, alterando a sequência de instruções a executar.

O desvio pode ser absoluto para endereços múltiplos de 4. O ajuste dos rótulos é feito pela pseudo-instrução BND.

O desvio pode ser relativo compreendido entre -128 e +127.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	JDU						‹rótulo›
	JDS						‹rótulo›
	JRU						‹rótulo›
	JRS						‹rótulo›

Observações:
JDU - desvio absoluto.
JDS - desvio absoluto para subrotina.
JRU - desvio relativo.
JRS - desvio relativo para subrotina.

- Instruções de teste e desvio:

Estas instruções testam uma determinada condição

e, de acordo com o resultado altera ou não o valor do MPC.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	TST				bt		⟨registrador⟩,⟨rótulo⟩
	SKP				bxtx	byty	⟨registrador⟩

Observações:

TST - desvia para a linha de programa com o rótulo especificado, se o bit do registrador testado for igual ao valor a testar.

SKP - não executa a instrução seguinte quando para o registrador usado, os seus dois bits especificados corresponderem aos dois valores testados.

b - número do bit a testar, de 0 a F.

t - valor a testar, 0 ou 1.

x - valor a testar, 0 ou 1.

y - valor a testar, 0 ou 1.

- Instruções de entrada e saída:

Estas instruções executam a entrada e saída de dados entre a UCP e os canais do sistema ED 311. Maiores detalhes podem ser encontrados em /PAD 80/.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	RD		BFC		⊮	⊮	⟨função barram. R⟩
			IOA		CN	AC	
					CF	RQ	
	WT		BFC		⊮	⊮	⟨função barram. R⟩
			IOA		CN	AC	
					CF	RQ	

- Instruções de referência à memória de controle:

Estas intruções fazem a transferência entre o registrador D e uma palavra da memória de controle cujo endereço está no registrador A.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	CMRD						
	CMWT						

Observações:
CMRD - leitura.
CMWT - escrita.

- Instruções aritméticas e lógicas:

Estas instruções definem as operaçöes de aritmética e lógica a serem executadas pelo microprocessador. A unidade aritmética e lógica sempre utiliza os registradores B e D como operandos.

Após a operação de aritmética e lógica, poderá ser feita uma operação de acesso à memória principal ou esperar pelo fim da operação de acesso à memória principal.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	MAR	Ø	+@	Ø			<registrador>
		R	AND	0-3			
		W	EOR	0-C			
		&	OR	8-C			
			+0				
			+1				
			-1				
			+S				
	MAG	Ø	+@	Ø			<reg. geral>
		R	AND	0-3			
		W	EOR	0-C			
		&	OR	8-C			
			+0				
			+1				
			-1				
			+S				

Observações:

Campo & : especifica o tipo de acesso à MP que deve ser realizado após executar a operação.

Ø : não faz acesso à MP.

R : lê o conteúdo para D.

W : grava o conteúdo de D.

& : espera o fim da operação de acesso à MP.

Campo M1 : especifica a operação a executar com os registradores B e D.

+@ : adição decimal.

AND : E lógico.

EOR : OU exclusivo.

+0 : adição binária.

- Instruções de referência a registrador:

Estas instruções manipulam dados de registradores, alterando seu conteúdo ou movendo-o para outro registrador. São divididas em instruções que ainda podem fazer um acesso à memória principal e as que não realizam este acesso. Maiores detalhes a respeito do significado de cada uma das instruções podem ser encontrados em /PAD 80/.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	MID						⟨dado⟩,⟨dest.⟩
	SET						⟨b8⟩,⟨b9⟩,...,⟨b15⟩
	RSET						⟨b8⟩,⟨b9⟩,...,⟨b15⟩
	MIN						⟨orig.⟩
	MOUT						⟨dest.⟩
	MILM						⟨ender. da LM⟩
	MOLM						⟨ender. da LM⟩
	NOP						
	HLT						⟨código⟩

Observações:

Estas instruções não acessam a memória principal.

MID - move um valor de 12 bits para os bits de 4 a 15 do registrador especificado. Os bits de 0 a 3 são zerados.

SET - liga os bits de 8 a 15 do registrador S.

RSET - desliga os bits de 8 a 15 do registrador S.

MIN - move dado do registrador especificado para o registrador D.

MOUT - move dado do registrador D para o registrador especificado.

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
	MURR	Ø	+1				⟨orig.⟩,⟨dest.⟩
		R	-1				
		W	+S				
		&	-S				
	MURG	Ø	+1				⟨orig.⟩,⟨reg. geral⟩
		R	-1				
		W	+S				
		&	-S				
	MDRR	Ø	INV	Ø			⟨orig.⟩,⟨dest.⟩
		R	4	X			
		W	8	XSL			
		&	C	XSR			
	MDRG	Ø	INV	Ø			⟨orig.⟩,⟨reg. geral⟩
		R	4	X			
		W	8	XSL			
		&	C	XSR			
	MDGR	Ø	INV	Ø			⟨reg. geral⟩,⟨dest.⟩
		R	4	X			
		W	8	XSL			
		&	C	XSR			
	MSRR	Ø	Ø				⟨orig.⟩,⟨dest.⟩
		R	X				
		W	XSL				
		&	XSR				
			SL				
			SR				
			LSL				
			LSR				
	MSRG	Ø	Ø				⟨orig.⟩,⟨reg. geral⟩
		R	X				
		W	XSL				
		&	XSR				
			SL				

ROT.	CÓDIGO	&	M1	M2	C1	C2	OPERANDOS
			SR				
			LSL				
			LSR				
	MSGR	Ø	Ø				<reg.geral>,<dest.>
		R	X				
		W	XSL				
		&	XSR				
			SL				
			SR				
			LSL				
			LSR				

Observações:

Campo M1 : especifica o tipo de incremento ou decremento a ser realizado no valor do registrador de origem antes de ser movido para o registrador destino.

+1 : soma 1, guarda "vai-um" no bit 3 de S.

-1 : subtrai 1, guarda "vai um" no bit 3 de S.

+S : se bit 3 de S é igual a 1, soma 1 ao registrador origem e guarda "vai-um" no bit 3 de S.

Campo M2 e M1 : especificam respectivamente, a primeira e a segunda operação a ser realizada antes da transferência para o registrador destino.

INV : complemento.

4 : zera bits de 0 a 11.

8 : zera bits de 0 a 7.

C : zera bits de 0 a 3.

Ø : não desloca bits.

X : troca bytes.

XSL : rotação à esquerda de 4 bits.

XSR : rotação à direita de 4 bits.

Campo M1 : indica o deslocamento a realizar.

- Ø : não altera o conteúdo.
- X : troca bytes.
- XSL : rotação à esquerda de 4 bits.
- XSR : rotação à direita de 4 bits.
- SL : deslocamento à esquerda de 1 bit.
- SR : deslocamento à direita de 1 bit.
- LSL : rotação à esquerda de 1 bit.
- LSR : rotação à direita de 1 bit.

ANEXO 2

MDE (Microdepurador Editor para o ED 311)

Os comandos do MDE utilizados nos testes do emulador, gerente e carregador estão listados a seguir, conforme a função, com o seu significado e formato.

- Comandos de entrada e saída:

a) **CF**, trilha, setor, num. de setores, endereço
Carga na memória principal a partir de disco Elexível.

b) **CD**, trilha, setor, num. de setores, endereço
Carga na memória principal a partir de Disco rígido.

c) **DF**, trilha, setor, num. de setores, endereço
Descarga em disco Elexível de conteúdo da memória principal.

d) **DD**, trilha, setor, num. de setores, endereço
Descarga em Disco rígido de conteúdo da memória principal.

e) **IM**, endereço, tamanho
Impressão do conteúdo da Memória principal em hexadecimal.

f) **IC**, endereço, tamanho
Impressão do conteúdo da memória de Controle em hexadecimal.

- Comandos de transferência:

a) TM, endereço fonte, endereço dest., tamanho
Iransferência a partir da memória de controle para a Memória principal.

b) TC, endereço fonte, endereço dest., tamanho
Iransferência a partir da memória principal para a memória de Controle.

- Comandos de edição (verificação) pelo terminal:

a) VM, endereço, tamanho
Verificação do conteúdo da Memória principal.

b) VC, endereço, tamanho
Verificação do conteúdo da memória de Controle.

c) VR
Verificação do conteúdo de todos os Registradores da UCP.

d) VR, mnemônico, mnemônico, ...
Verificação de conteúdo de Registrador da UCP identificado.

e) VP, número de posições
Verificação de conteúdo de posições da Pilha da UCP.

f) VG

Verificação do conteúdo de todos os registradores Gerais.

g) VG, identificador, identificador, ...
Verificação do conteúdo dos registradores Gerais identificados.

h) VL, posição, tamanho
Verificação do conteúdo da memória Local (posição 0 a 10).

i) VT

Verificação do conteúdo da ILB.

- Comandos de edição (alteração) pelo terminal:

a) AM, endereço, tamanho
Alteração do conteúdo da Memória principal.

b) AC, endereço, tamanho
Alteração do conteúdo da memória de Controle.

c) AR, mnemônico
Alteração do conteúdo de Registrador da UCP.

d) AP, número de posições
Alteração do conteúdo de posições da Pilha da UCP.

e) AG, identificador

Alteração do conteúdo de registrador Geral.

f)AL, posição, tamanho

Alteração do conteúdo da memória Local.

g)AT, identificador

Alteração do conteúdo da ILB.

- Comandos de depuração:

a) PP, endereço

inserção de Ponto de Parada (no máximo 8) no microprograma sob teste que se encontra na memória de controle.

b) SP, endereço

Supressão de ponto de Parada.

c) SP

Supressão de todos os pontos de Parada.

d) LP

Lista no terminal todos os pontos de Parada inseridos no microprograma sob teste.

- Comandos de execução:

a) EX, endereço

Execução do microprograma sob teste a partir de um determinado endereço.

b) EX

Execução do microprograma sob teste a partir do último ponto de parada.

- Procedimentos para depuração de um microprograma:

a) Carregar o MDE na memória de controle a partir de disco flexível (IMPL pelo painel).

b) Com o MDE carregado, utilizando o terminal, usar os comandos de entrada (CF ou CD) para carregar o microprograma a ser testado na memória principal.

c) Transferir o microprograma sob teste da memória principal para a memória de controle onde o microprograma será executado (comando TC).

d) Usar os comandos de edição e depuração para teste do microprograma.

e) Transferir o microprograma depurado da memória de controle para a memória principal (comando TM).

f) Descarregar o microprograma depurado em disco flexível ou disco (DF ou DD) a partir da memória principal.

ANEXO 3

FORMA DE APRESENTAÇÃO DOS ALGORITMOS

Os algoritmos, neste trabalho, estão apresentados em uma linguagem bloco estruturada semelhante ao Pascal, onde os comandos foram traduzidos para português.

Esta linguagem oferece flexibilidade para que se possa descrever livremente em lingua portuguesa, a estratégia desejada.

Alguns tópicos exigem explicações:

- A atribuição utiliza o símbolo <---.

- Os comentários aparecem em letras maiúsculas, entre dois asteriscos.

- As variáveis utilizadas são apresentadas no início do algoritmo antecedidas por um asterisco, com sua função descrita em português.

- Para chamar um procedimento (subrotina) usa-se o nome do procedimento em letras maiúsculas e os parâmetros,se houverem, entre parênteses e separados por vírgula.

  Os algoritmos dos procedimentos invocados são apresentados logo após o algoritmo principal.

- Para se referir às tabelas e arquivos usa-se o nome por extenso conforme especificado no texto ou a abreviatura que consta da lista de abreviaturas.

- Para se referir aos campos das tabelas e arquivos usa-se o seguinte formato:

  [nome da tabela].[nome do campo]

- Para se referir aos bits de uma palavra de memória usa-se o seguinte formato:

  [endereço / nome] [(b$_n$: b$_m$)]

  b$_n$ é o número do bit onde inicia o campo.

  b$_m$ é o número do bit onde termina o campo.

Alguns procedimentos, para facilitar a visualização, são apresentados em forma de diagramas de blocos (fluxogramas), utilizando a simbologia tradicional.

ANEXO 4

1%

```
EDISA  ED330 * MONTADOR DE MICROPROGRAMAS *  MICMON 534-4007                                          ***(MICROT  )***       PAG 0001 86/11/10

                                                           C  .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG  DEST  LOC   INST  CONFIGURACAO         BITS  NINS  ROTULO OPER S M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO                ID/SEQ
                                                   0001          ELM                  EMULPCIR                                    EMU00010
            0880                                   0002          ORG                  X'880'                                      EMU00020
                                                   0003          JDS                  IDENT                                       EMU00030
            0880                                   0004          DC                   PCIR1                                       EMU00040
      2020  0880  A808  1010 1000 0000 1000        0005          DC                   PCIR2                                       EMU00050
      0E44  0881  A391  1010 0011 1001 0001        0006          JDS                  TVINIC                                      EMU00060
095C        0882  095C  0000 1001 0101 1100        0007 *----------------------------GERENCIADOR*-----------------------------------EMU00070
                                                   0008          BND                                                              EMU00080
0008  D     0883  CC22  1100 0000 0010 0010        0009  PARAR   JDS                  STKCOM                                      EMU00090
D     GR1   0884  21C1  0010 0001 1100 0001        0010          JDS                  IMPVID                                      EMU00100
      0A00  0885  A280  1010 0010 1000 0000        0011          DC                   PCIR1                                       EMU00110
                                                   0012 *PREPARA PARAMETROS PARA CARREGAR O PRIMEIRO SETOR DO DISQ. MP FF60          EMU00120
      2000  0886  A800  1010 1000 0000 0000        0013          MID                  X'008',D          D=TRILHA 0 SETOR1         EMU00130
08C6        0887  08C6  0000 1000 1100 0110        0014          MSRG                 D,GR1                                       EMU00140
A     GR1   0888  2241  0010 0010 0100 0001        0015          JDS                  CF                MP FF60 <--DISQ 0 ,TRI 1  EMU00150
      2000  0889  A800  1010 1000 0000 0000        0016 *PREPARA PARAMETROS PARA TRANSFERIR MP P/ MC CHAMANDO ROTINA "TRLPC"      EMU00160
08C7        088A  08C7  0000 1000 1100 0111        0017          JDS                  LDAI                                        EMU00170
A     GR0   088B  2240  0010 0010 0100 0000        0018          DC                   ENDMP                                       EMU00180
A     D     088C  3203  0011 0010 0000 0011        0019          MSRG                 A,GR1                                       EMU00190
      2008  088D  A808  1010 1000 0000 1000        0020          JDS                  LDAI                                        EMU00200
0907        088E  0907  0000 1001 0000 0111        0021          DC                   AREACG                                      EMU00210
0040  D     088F  C102  1100 0001 0000 0010        0022          MSRG                 A,GR0                                       EMU00220
D     D     0890  3102  0011 0001 0000 0010        0023          MSRR                 A,D                                         EMU00230
            0891  0815  0000 1000 0001 0101        0024          JDS                  SDD                                         EMU00240
      CF39  0892  A39E  1010 0011 1001 1110        0025          DC                   CONTRI            CONTRI=CONTADOR DE WORDS  EMU00250
                                                   0026          MID                  64,D                                        EMU00260
      0E44  0893  A391  1010 0011 1001 0001        0027          MSRR                 D,B                                         EMU00270
0E60        0894  0E60  0000 1110 0110 0000        0028          RSET                 14                                          EMU00280
      2014  0895  A805  1010 1000 0000 0101        0029          JDS                  TRLPC             MC AREACG <--MP FF60      EMU00290
0907        0896  0907  0000 1001 0000 0111        0030 *INICIO DO GERENCIADOR PROPRIAMENTE DITO------------------------------------EMU00300
D     A     0897  3144  0011 0001 0100 0100        0031  INIGER  JDS                  IMPVID                                      EMU00310
            0898  0803  0000 1000 0000 0011        0032          DC                   MENSA1                                      EMU00320
D           0899  [illegible]  1000 1101 1000 0000 0033  GGGG    JDS                  LDD                                         EMU00330
      [illegible] 089A 3221 [illegible]            0034          DC                   CONTRI            D<--ENDERECO              EMU00340
      2000  089B  A800  1010 1000 0000 0000        0035          MSRR                 D,A                                         EMU00350
0909        089C  0909  0000 1001 0000 1001        0036          CHRD                                   D<--PRIM. PALAVRA         EMU00360
            089D  0902  0000 1000 0000 0010        0037          SKP            01 01 D                                           EMU00370
D     GR1   089E  21C1  0010 0001 1100 0001        0038          JDU                  PARAR             PARAR A MAQUINA PCIR      EMU00380
      0CF3  089F  A33E  1010 0011 0011 1110        0039          JDS                  LDAI.                                       EMU00390
      2014  08A0  A805  1010 1000 0000 0101        0040          DC                   WORD1                                       EMU00400
0D04        08A1  0D04  0000 1101 0000 0100        0041          CMWT                                   WORD1<--D                 EMU00410
D     GR3   08A2  21C3  0010 0001 1100 0011        0042          MSRG                 D,GR1                                       EMU00420
                                                   0043          JDS                  FETPAR            TRIL1=TRILHA SET1=SETOR   EMU00430
      2014  08A3  A805  1010 1000 0000 0101        0044          JDS                  LDD                                         EMU00440
0D05        08A4  0D05  0000 1101 0000 0101        0045          DC                   TRIL1                                       EMU00450
D     GR0   08A5  21C0  0010 0001 1100 0000        0046          MSRG                 D,GR3             PARAMETRO P/ CARREGADOR   EMU00460
                                                   0047 *                                               GR3=TRILHA INICIAL        EMU00470
      2014  08A6  A805  1010 1000 0000 0101        0048          JDS                  LDD                                         EMU00480
0907        08A7  0907  0000 1001 0000 0111        0049          DC                   SET2                                        EMU00490
D     A     08A8  11E4  0001 0001 1110 0100        0050          MSRG                 D,GR0             PARAMETRO P/CARREGADOR    EMU00500
            08A9  0803  0000 1000 0000 0011        0051 *                                               GR0=SETOR INICIAL         EMU00510
D     GR1   08AA  21C1  0010 0001 1100 0001        0052          JDS                  LDD                                         EMU00520
                                                   0053          DC                   CONTRI                                      EMU00530
A     A     08AB  1264  0001 0010 0110 0100        0054          MURR   +1            D,A                                         EMU00540
            08AC  0803  0000 1000 0000 0011        0055          CHRD                                   D<--SEG. PALAVRA          EMU00550
D     GR2   08AD  21C2  0010 0001 1100 0010        0056          MSRG                 D,GR1             PARAMETRO P/ CARREGADOR   EMU00560
                                                   0057 *                                               GR1= NUMERO DE SETORES    EMU00570
D     0881  08AE  93F2  1001 0011 1111 0010        0058          MURR   +1            A,A                                         EMU00580
```

197

```
ORIG DEST LOCN INST CONFIGURACAO        BITS  NINS  C . 1 . 2 . 3 . 4 . 5 . 6 . 7 . 8
                                                   ROTULO OPER S M1/M2 C1 C2 ORIGEM/DESTINO   COMENTARIO                    ID/SEQ
     2008 08AF A802 1010 1000 0000 0010     0059          JDS              SDD             IPARAMETRO PARA O EMULADORIEMU00590
0908      08B0 0908 0000 1001 0000 1000     0060          DC               CAREND          INICIALIZAR O PC             IEMU00600
A    D    08B1 1263 0001 0010 0110 0011     0061          MURG +1          A,D                                           EMU00610
     2008 08B2 A802 1010 1000 0000 0010     0062          JDS              SDD                                           EMU00620
0907      08B3 0907 0000 1001 0000 0111     0063          DC               CONTRI                                        EMU00630
     090C 08B4 A243 1010 0010 0100 0011     0064          JDS              LOADER                                        EMU00640
                                            0065   *-----QUANDO VOLTAR DO CARREGADOR:                                    EMU00650
     2014 08B5 A805 1010 1000 0000 0101     0066          JDS              LDD                                           EMU00660
0909      08B6 0909 0000 1001 0000 1001     0067          DC               WORD1                                         EMU00670
D         08B7 8FDD 1000 1111 1101 1101     0068          SKP        01 01 D               SE BEX=1 ENTAO EXECUTAR       EMU00680
     0895 08B8 06E3 0000 0110 1110 0011     0069          JRU              GGGG            SENAO CARGA                   EMU00690
     0E44 08B9 A391 1010 0011 1001 0001     0070          JDS              IMPVID                                        EMU00700
0E64      08BA 0E64 0000 1110 0110 0100     0071          DC               MENSA2                                        EMU00710
     0E44 08BB A391 1010 0011 1001 0001     0072          JDS              IMPVID                                        EMU00720
0E68      08BC 0E68 0000 1110 0110 1000     0073          DC               MENSA3                                        EMU00730
     0922 08BD A24A 1010 0010 0100 1010     0074          JDS              PCIN16                                        EMU00740
0908      08BE 0908 0000 1001 0000 1000     0075          DC               CAREND          ENDERECO ONDE O MODULO FOI    EMU00750
                                            0076   *                                 CARREGADO. VALOR INICIAL DO PC      EMU00760
                                            0077   *----QUANDO VOLTAR DO EMULADOR:                                       EMU00770
     0E44 08BF A391 1010 0011 1001 0001     0078          JDS              IMPVID                                        EMU00780
0E6C      08C0 0E6C 0000 1110 0110 1100     0079          DC               MENSA4                                        EMU00790
     2014 08C1 A805 1010 1000 0000 0101     0080          JDS              LDD                                           EMU00800
0909      08C2 0909 0000 1001 0000 1001     0081          DC               WORD1                                         EMU00810
D    08C5 08C3 93E1 1001 0011 1110 0001     0082          TST        EQ    D,*+2                                         EMU00820
     0D14 08C4 A345 1010 0011 0100 0101     0083          JDS              ESTATI                                        EMU00830
     0893 08C5 066B 0000 0110 0110 1011     0084          JRU              INIGER                                        EMU00840
                                            0085   *--------------VARIAVEIS DO GERENCIADOR---------------------------------EMU00850
FF60      08C6 FF60 1111 1111 0110 0000     0086   ENOMP  DC               X'FF60'                                       EMU00860
          08C7                              0087   AREACG DA               64                                            EMU00870
          0907                              0088   CONTRI DA               1                                             EMU00880
          0908                              0089   CAREND DA               1                                             EMU00890
          0909                              0090   WORD1  DA               1                                             EMU00900
                                            0091   *------------------FIM DO GERENCIADOR-----------------------------------EMU00910
                                            0092   *---------------------CARREGADOR----------------------------------------EMU00920
                                            0093   *---GR0=SETOR INICIAL                                                 EMU00930
                                            0094   *---GR1=NUMERO DE SETORES                                             EMU00940
                                            0095   *---GR2=ENDERECO DE CARGA NA MEMORIA PRINCIPAL                        EMU00950
                                            0096   *---GR3=TRILHA INICIAL, PARAMETRO DA ROTINA "FDRD"                    EMU00960
                                            0097   *---GR4=FFD,PARAMETRO DA ROTINA FDRD                                  EMU00970
                                            0098   *---GR5=1, PARAMETRO DA ROTINA "FDRD"                                 EMU00980
                                            0099   *-----------------------------------------------------------------------EMU00990
          090C                              0100          END                                                            EMU01000
0001 D    090C C006 1100 0000 0000 0110     0101   LOADER MID              1,D                                           EMU01010
     2008 090D A802 1010 1000 0000 0010     0102          JDS              SDD                                           EMU01020
0AD0      090E 0AD0 0000 1010 1101 0000     0103          DC               FDFLAG                                        EMU01030
GR2  D    090F 2103 0010 0001 0000 0011     0104   FDCAR  MSGR             GR2,D                                         EMU01040
D    GR6  0910 21C6 0010 0001 1100 0110     0105          MSRG             D,GR6                                         EMU01050
0080 B    0911 C201 1100 0010 0000 0001     0106          MID              128,B                                         EMU01060
     B    0912 1232 0001 0010 0011 0010     0107          MAR   +0 0-C     B                                             EMU01070
B    GR2  0913 2142 0010 0001 0100 0010     0108          MSRG             B,GR2                                         EMU01080
GR0  D    0914 2003 0010 0000 0000 0011     0109          MSGR             GR0,D                                         EMU01090
D    GR4  0915 21C4 0010 0001 1100 0100     0110          MSRG             D,GR4                                         EMU01100
D    GR0  0916 11C0 0001 0001 1100 0000     0111          MURG  +1         D,GR0                                         EMU01110
0001 D    0917 C006 1100 0000 0000 0110     0112          MID              1,D                                           EMU01120
D    GR5  0918 21C5 0010 0001 1100 0101     0113          MSRG             D,GR5                                         EMU01130
     2870 0919 AA1C 1010 1010 0001 1100     0114          JDS              FDRD            LE DISQUETE                   EMU01140
GR1  D    091A 2083 0010 0000 1000 0011     0115          MSGR             GR1,D                                         EMU01150
D    GR1  091B 11C9 0001 0001 1100 1001     0116          MURG  -1         D,GR1                                         EMU01160
```

```
EDISA  ED300 * MONTADOR DE MICROPROGRAMAS *  MICRON E24-R000                        ***(MICROT  )***      PAG 0003 86/11/10

                                                    C  .  1  .  2   .  3   .  4   .  5   .   6   .   7   .  8
ORIG DEST LOCN INST CONFIGURACAO        NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                ID/SEQ
S    0926 091C 9079 1001 0000 0111 1001 0117         TST          70    S,FDFIMM                                  EMU01170
0015 B    091D C060 1100 0000 0110 1101 0118         MID                27,B                                      EMU01180
GR0  D    091E 2003 0010 0000 0000 0011 0119         MSGR               GR0,D                                     EMU01190
     B    091F 1122 0001 0001 0010 0010 0120         MAR     EOR        B                                         EMU01200
S    0925 0920 9474 1001 0100 0111 0100 0121         TST          71    S,FONMUD                                  EMU01210
0001 D    0921 C006 1100 0000 0000 0110 0122         MID                1,D                                       EMU01220
D    GR0  0922 21C0 0010 0001 1100 0000 0123         MSRG               D,GR0                                     EMU01230
GR3  D    0923 2133 0010 0001 1000 0011 0124         MSGR               GR3,D                                     EMU01240
D    GR3  0924 11C3 0001 0001 1100 0011 0125         MORG    +1         D,GR3                                     EMU01250
     090F 0925 0743 0000 0111 0100 1011 0126  FONMUD JRU                FDCAR                                     EMU01260
STK  MPC  0926 3307 0011 0011 0000 0111 0127  FDFIMM MSRR               STK,MPC         TERMINOU A CARGA          EMU01270
                                        0128  *------------------FIM DO CARREGADOR--------------------------------EMU01280
          0928                          0129         RND                                                          EMU01290
     2014 0928 A805 1010 1000 0000 0101 0130  PCIR16 JDS                LDD                                       EMU01300
0909      0929 0909 0000 1001 0000 1001 0131         DC                 +ORG1                                     EMU01310
D         092A 83EE 1000 0011 1110 1110 0132         SKP          E1 E0 D                                         EMU01320
     0008 092B A342 1010 0011 0100 0010 0133         JDS                ZERACT                                    EMU01330
                                        0134  *-----------------LM(0)<--ENDERECO BASE DA TABELA DE REGISTRADORES------EMU01340
00E0 D    092C C382 1100 0011 1000 0010 0135         MID                X'0E0',D        D<--'00E0'                EMU01350
00FF B    092D C3FD 1100 0011 1111 1101 0136         MID                X'0FF',B        B<--'00FF'                EMU01360
B    B    092E 310A 0011 0001 0000 1010 0137         MSRR    X          B,B             B<--'FF00'                EMU01370
     D    092F 1223 0001 0010 0010 0011 0138         MAR     +0         D               D<--'FFE0'                EMU01380
     LM0  0930 0005 0000 0000 0000 0101 0139         MDLM               0                LM(0)<--'FFE0'           EMU01390
                                        0140  *---------------------INICIALIZAR PC------------------------------- EMU01400
STK  A    0931 3304 0011 0011 0000 0100 0141         MSRR               STK,A                                     EMU01410
A    D    0932 1263 0001 0010 0110 0011 0142         MORR    +1         A,D                                       EMU01420
D    STK  0933 3186 0011 0001 1000 0110 0143         MSRR               D,STK                                     EMU01430
          0934 0803 0000 1000 0000 0011 0144         CMRD                               D<--ENDER INICIO PROG OBJ EMU01440
     LM1  0935 000D 0000 0000 0000 1101 0145         MDLM               1               LM(1)<--'0000'            EMU01450
000F B    0936 C03D 1100 0000 0011 1101 0146         MID                X'00F',B        EM FASE DE TESTE          EMU01460
     0A50 0937 A284 1010 0010 1000 0100 0147         JDS                GRAVRG          PC <----D                 EMU01470
     0A30 0938 A28C 1010 0010 1000 1100 0148         JDS                PREFET          BUSCA PRIM. INSTRUCAO     EMU01480
     093C 0939 B24F 1011 0010 0100 1111 0149         JDU                TESTE                                     EMU01490
          093C                          0150         RND                                                          EMU01500
                                        0151  *-------------------------TESTE-------------------------------------EMU01510
                                        0152  *----LEITURA DA PALAVRA FFSC, COLOCA CONTEUDO EM GR1----------------EMU01520
     2014 093C A805 1010 1000 0000 0101 0153  TESTE  JDS                LDD                                       EMU01530
095A      093D 095A 0000 1001 0101 1010 0154         DC                 ENDPAR                                    EMU01540
D    A    093E 3184 0011 0001 1000 0100 0155         MSRR               D,A                                       EMU01550
          093F 0816 0000 1000 0001 0110 0156         RSET               14                                        EMU01560
B    B    0940 3502 0011 0101 0000 0010 0157         MSRR R             B,B             D<--MEMP FFSC             EMU01570
B    B    0941 3D02 0011 1101 0000 0010 0158         MSRR &             B,B                                       EMU01580
D    GR1  0942 21C1 0010 0001 1100 0001 0159         MSRG               D,GR1                                     EMU01590
                                        0160  *---TESTA BIT DE EXECUCAO DE IO-------------------------------------EMU01600
D    0945 0943 97F1 1001 0111 1111 0001 0161         TST          F1    D,ENTSAI                                  EMU01610
     0A44 0944 B291 1011 0010 1001 0001 0162         JDU                FETCH                                     EMU01620
     2014 0945 A805 1010 1000 0000 0101 0163  ENTSAI JDS                LDD             I-------------------------EMU01630
095A      0946 095A 0000 1001 0101 1010 0164         DC                 ENDPAR          I                         EMU01640
D    A    0947 3184 0011 0001 1000 0100 0165         MSRR               D,A             I                         EMU01650
0000 D    0948 C002 1100 0000 0000 0010 0166         MID                0,D             I                         EMU01660
          0949 0816 0000 1000 0001 0110 0167         RSET               14              I ZERA BIT DE EXECUCAO I/O EMU01670
B    B    094A 3902 0011 1001 0000 0010 0168         MSRR *             B,B             I                         EMU01680
B    B    094B 3D02 0011 1101 0000 0010 0169         MSRR &             B,B             I------------------------ EMU01690
GR0  D    094C 2003 0010 0000 0000 0011 0170         MSGR               GR0,D           I------------------------ EMU01700
     2008 094D A802 1010 1000 0000 0010 0171         JDS                SDD             I  SALVA GR0              EMU01710
0958      094E 0958 0000 1001 0101 1011 0172         DC                 SLVGR0          I-------------------------EMU01720
                                        0173  *---DECODIFICA INSTRUCAO DE I/O (BITS 13 A 15)----------------------EMU01730
0007 D    094F C01E 1100 0000 0001 1110 0174         MID                X'007',D                                  EMU01740
```

```
ORIG DEST LOCN INST CONFIGURACAO        NINS  C . 1 . 2 . 3 . 4 . 5 . 6 . 7 . 8
                                              ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO        ID/SEQ
GR1  B    0950 2082 0010 0000 1000 0010 0175         MSGR             GR1,B                               EMU01750
     B    0951 10A2 0001 0000 1010 0010 0176         MAR   AND        B                                   EMU01760
0958 D    0952 E556 1110 0101 0101 0110 0177         MID              TABIO,D                             EMU01770
     D    0953 1223 0001 0010 0010 0011 0178         MAR   +0         D                                   EMU01780
D    MPC  0954 3187 0011 0001 1000 0111 0179         MSRR             D,MPC                               EMU01790
     0A00 0955 B280 1011 0010 1000 0000 0180  TABIO  JDU              CF                                  EMU01800
     0A08 0956 B282 1011 0010 1000 0010 0181         JDU              DF                                  EMU01810
     09A8 0957 B26A 1011 0010 0110 1010 0182         JDU              LINIMP                              EMU01820
     09F8 0958 B27E 1011 0010 0111 1110 0183         JDU              LERVID                              EMU01830
     09FC 0959 B27F 1011 0010 0111 1111 0184         JDU              GRAVID                              EMU01840
FF5C      095A FF5C 1111 1111 0101 1100 0185  ENDPAR DC               X'FF5C'                             EMU01850
          095B                          0186  SLVGRO DA               1                                   EMU01860
                                        0187 ************************R O T I N A S  DE E N T R A D A  E  S A I D A****EMU01870
          095C                          0188         BND                                                  EMU01880
2020      095C 2020 0010 0000 0010 0000 0189  PCIR1  DC               X'2020'                             EMU01890
4D41      095D 4D41 0100 1101 0100 0001 0190         DC               X'4D41'         'MA'                EMU01900
5155      095E 5155 0101 0001 0101 0101 0191         DC               X'5155'         'QU'                EMU01910
494E      095F 494E 0100 1001 0100 1110 0192         DC               X'494E'         'IN'                EMU01920
4120      0960 4120 0100 0001 0010 0000 0193         DC               X'4120'         'A '                EMU01930
5043      0961 5043 0101 0000 0100 0011 0194  PCIR2  DC               X'5043'         'PC'                EMU01940
4952      0962 4952 0100 1001 0101 0010 0195         DC               X'4952'         'IR'                EMU01950
2031      0963 2031 0010 0000 0011 0001 0196         DC               X'2031'         ' 1'                EMU01960
3620      0964 3620 0011 0110 0010 0000 0197         DC               X'3620'         '6 '                EMU01970
2020      0965 2020 0010 0000 0010 0000 0198         DC               X'2020'                             EMU01980
0A00      0966 0A00 0000 1010 0000 0000 0199         DC               X'0A00'                             EMU01990
                                        0200 ***ESTA ROTINA TRANSFERE OS DADOS DA AREA DE DADOS DA MP PARA A AREA DE EMU02000
                                        0201 *NOME "DADOS" NA MEMORIA DE CONTROLE*********************************** EMU02010
          0967                          0202  DADOS  DA               64                                  EMU02020
7BAF      09A7 7BAF 0111 1011 1010 1111 0203  ENDIO  DC               X'7BAF'                             EMU02030
          09A8                          0204         BND                                                  EMU02040
0001 D    09A8 C002 1100 0000 0000 0010 0205  LINIMP MID              X'001',D                            EMU02050
D    GR3  09A9 21C3 0010 0001 1100 0011 0206         MSRR             D,GR3                               EMU02060
     2014 09AA A805 1010 1000 0000 0101 0207         JDS              LDD             ZERA PONTEIRO DA AREA DADOSEMU02070
09A7      09AB 09A7 0000 1001 1010 0111 0208         DC               ENDIO                               EMU02080
D    A    09AC 11E4 0001 0001 1110 0100 0209  LPIO   MURR  +1         D,A             D<---7BAF           EMU02090
A    GR4  09AD 2244 0010 0010 0100 0100 0210         MSRR             A,GR4           A<--ENDER MP ATUALIZADO EMU02100
          09AE 0000 0000 0000 0000 0000 0211         NOP                              GR4<--GUARDAEND MP  EMU02110
     0A4C 09AF A293 1010 0010 1001 0011 0212         JDS              AJENDR                              EMU02120
          09B0 0816 0000 1000 0001 0110 0213         RSET             14              A<--ENDERECO AJUSTADO EMU02130
B    B    09B1 3502 0011 0101 0000 0010 0214         MSRR  B          B,A                                 EMU02140
B    B    09B2 3D02 0011 1101 0000 0010 0215         MSRR  B          A,B                                 EMU02150
D    GR5  09B3 21C5 0010 0001 1100 0101 0216         MSRR             D,GR5           D<--DADO I/O    ----EMU02160
GR3  B    09B4 2182 0010 0001 1000 0010 0217         MSGR             GR3,B           GR5<--DADO I/O------EMU02170
     2014 09B5 A805 1010 1000 0000 0101 0218         JDS              LDAI              B<--PONTEIRO AREA DADOS-EMU02180
0967      09B6 0967 0000 1001 0110 0111 0219         DC               DADOS                               EMU02190
     A    09B7 1224 0001 0010 0010 0100 0220         MAR   +0         A                 D<--DADOS         EMU02200
GR5  D    09B8 2283 0010 0010 1000 0011 0221         MSGR             GR5,D             A<--ENDER M.CONTROLE EMU02210
          09B9 0802 0000 1000 0000 0010 0222         CHWT                               D<--DADO I/O------EMU02220
GR3  B    09BA 2182 0010 0001 1000 0010 0223         MSGR             GR3,B           MC A <-DADO         EMU02230
GR4  D    09BB 2203 0010 0010 0000 0011 0224         MSGR             GR4,D                               EMU02240
B    B    09BC 1172 0001 0001 0111 0010 0225         MURR  +1         B,B             DEVOLVE ENDERECO M.P. EMU02250
B    GR3  09BD 2143 0010 0001 0100 0011 0226         MSRR             B,GR3                               EMU02260
B    09C0 09BE 9711 1001 0111 0001 0001 0227         TST         91   B,IMPLIN         PONTAD<--PONTAD+1   EMU02270
     09AC 09BF B26B 1011 0010 0110 1011 0228         JDU              LPIO            SE ATINGIU 64 IR P/IMPLIN -EMU02280
                                        0229 ------------------------IMPRESSAO DE UMA LINHA SE NAO ATINGIU 64 VOLTA -EMU02290
     2000 09C0 A800 1010 1000 0000 0000 0230  IMPLIN JDS              LDAI                                -EMU02300
2521      09C1 2521 0010 0101 0010 0001 0231         DC               ASI                                 EMU02310
A    B    09C2 3202 0011 0010 0000 0010 0232         MSRR             A,B                                 EMU02320
```

```
ORIG DEST LOCN INST CONFIGURACAO        BITS NINS  C  .  1  .  2  .  3  .  4  .  5  .  6  .  7  .  8
                                                   ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO   COMENTARIO      ID/SEQ
0000 D    09C3 C002 1100 0000 0000 0010      0233         MID            0,0                               EMU02330
     GR2  09C4 1202 0001 0010 0000 0010      0234         MAG  +0        GR2                               EMU02340
     2023 09C5 A20A 1010 1000 0000 1010      0235         JOS            STKSLV                            EMU02350
090C      09C6 090C 0000 1001 0000 1100      0236         DC             ICPMEN                            EMU02360
0967      09C7 0967 0000 1001 0110 0111      0237         DC             DADOS                             EMU02370
     2514 09C8 A445 1010 0100 0100 0101      0238         JOS            PRINT                             EMU02380
     2014 09C9 A805 1010 1000 0000 0101      0239         JOS            LDD                               EMU02390
0953      09CA 0953 0000 1001 0101 0011      0240         DC             SLVGR0                            EMU02400
0    GR0  09CB 21C0 0010 0001 1100 0000      0241         MSRG           0,GR0           DEVOLVE GR0       EMU02410
     093C 09CC 324F 1011 0010 0100 1111      0242         JOU            TESTE                             EMU02420
                                             0243 ****************** R O T I N A  ICPMEN*********************EMU02430
                                             0244 *-----------GR2=POSICAO DA ASI A PARTIR DA QUAL SERA PREENCHIDA----------EMU02440
                                             0245 *-----------GR5=ENDERECO DA MENSAGEM------------------------------------EMU02450
                                             0246 *-----------GR6=MENSAGEM--------------------------------------------------EMU02460
          09D0                               0247         BND                                              EMU02470
     20A4 09D0 A829 1010 1000 0010 1001      0248  ICPMEN JOS            STKILE                            EMU02480
A    GR5  09D1 2245 0010 0010 0100 0101      0249         MSRG           A,GR5                             EMU02490
     20[illegible] 09D2 A[illegible] 1010 1000 0000 [illegible] 0250 JOS LDAI                              EMU02500
2521      09D3 2521 0010 0101 0010 0001      0251         DC             ASI                               EMU02510
0087 0    09D4 C210 1100 0011 0001 1101      0252         MID            135,3                             EMU02520
A    0    09D5 3203 0011 0010 0000 0011      0253         MSRG           A,0                               EMU02530
     GR3  09D6 1203 0001 0010 0000 0011      0254         MAG  +0        GR2             GR5-POS FINAL DA ASI EMU02540
     2023 09D7 A80A 1010 1000 0000 1010      0255         JOS            STKSLV                            EMU02550
09DA      09D8 09DA 0000 1001 1101 1010      0256         DC             ICPLAC                            EMU02560
     204C 09D9 B813 1011 1000 0001 0011      0257         JOU            STKREC                            EMU02570
GR5  A    09DA 2234 0010 0010 0011 0100      0258  ICPLAC MSGR           GR5,4                             EMU02580
A    GR5  09DB 1245 0001 0010 0100 0101      0259         MURG +1        A,GR5                             EMU02590
          09DC 0803 0000 1000 0000 0011      0260         CMRG                                             EMU02600
0    GR6  09DD 21C6 0010 0001 1100 0110      0261         MSRG           0,GR6           LE MENSAGEM       EMU02610
D    D    09DE 61CD 0110 0001 1100 1101      0262         MORG 2 Y       0,0                               EMU02620
09F7 6    09DF E7DD 1110 0111 1101 1101      0263         MID            NL,6                              EMU02630
     6    09E0 1122 0001 0001 0010 0010      0264         MAR  EOP       3                                 EMU02640
S    09E3 09E1 9471 1001 0100 0111 0001      0265         TST       71   S,*+2                             EMU02650
     09F6 09E2 0094 0000 0000 1001 0100     0266         JRU            ICPSAI                            EMU02660
GR2  A    09E3 2134 0010 0001 0011 0100      0267         MSGR           GR3,A                             EMU02670
          09E4 0802 0000 1000 0000 0010      0268         CMNT                                             EMU02680
A    0    09E5 1263 0001 0010 0110 0011      0269         MURG -1        A,0                               EMU02690
0    GR2  09E6 21C2 0010 0001 1100 0010      0270         MSRG           0,GR2                             EMU02700
GR3  0    09E7 2182 0010 0001 1000 0010      0271         MSGR           GR3,0                             EMU02710
     6    09E8 1122 0001 0001 0010 0010      0272         MAR  EOP       3                                 EMU02720
S    09F6 09E9 9070 1001 0000 0111 0000      0273         TST       70   S,ICPSAI                          EMU02730
GR6  0    09EA 4343 0100 0011 0100 0011      0274         MORG 8         GR6,0                             EMU02740
09F7 6    09EB E7DD 1110 0111 1101 1101      0275         MID            NL,6                              EMU02750
     6    09EC 1122 0001 0001 0010 0010      0276         MAR  EOF       3                                 EMU02760
S    09F6 09ED 9078 1001 0000 0111 1000      0277         TST       70   S,ICPSAI                          EMU02770
GR2  A    09EE 2134 0010 0001 0011 0100      0278         MSGR           GR2,A                             EMU02780
          09EF 0802 0000 1000 0000 0010      0279         CMNT                                             EMU02790
A    0    09F0 1263 0001 0010 0110 0011      0280         MURG +1        A,0                               EMU02800
0    GR2  09F1 21C2 0010 0001 1100 0010      0281         MSRG           0,GR2                             EMU02810
GR3  6    09F2 2182 0010 0001 1000 0010      0282         MSGR           GR3,3                             EMU02820
     6    09F3 1122 0001 0001 0010 0010      0283         MAR  EOP       0                                 EMU02830
S    09F6 09F4 9071 1001 0000 0111 0001      0284         TST       70   S,ICPSAI                          EMU02840
     09DA 09F5 0723 0000 0111 0010 0011      0285         JRU            ICPLAC                            EMU02850
     204C 09F6 B813 1011 1000 0001 0011      0286  ICPSAI JOU            STKREC                            EMU02860
000A      09F7 000A 0000 0000 0000 1010      0287  NL     DC             X'000A'                           EMU02870
          09F8                               0288 *-----------------------------LER VIDEO-------------------------------EMU02880
                                             0289         BND                                              EMU02890
     2014 09F8 A805 1010 1000 0000 0101      0290  LERVID JOS            LDD             I-------------------------IEMU02900
```

201


```
                                               C  .  1  .  2  .  3  .  4  .  5  .  6  .  7  .  8
ORIG DEST LOCN INST CONFIGURACAO   BITS NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO           ID/SEQ
0958      09F9 0958 0000 1001 0101 1011 0291         DC                  SLVGR0          I DEVOLVE GR0      IE4U02910
D    GR0  09FA 21C0 0010 0001 1100 0000 0292         MSRG                D,GR0           I------------------E4U02920
     093C 09FB B24F 1011 0010 0100 1111 0293         JDU                 TESTE                              E4U02930
                                        0294 *-----------------------ESCREVER NO VIDEO-------------------E4U02940
          09FC                          0295         BND                                                    E4U02950
     2014 09FC A805 1010 1000 0000 0101 0296  GRAVID JDS                 LDD             I------------------IE4U02960
0958      09FD 0958 0000 1001 0101 1011 0297         DC                  SLVGR0          I DEVOLVE GR0      IE4U02970
D    GR0  09FE 21C0 0010 0001 1100 0000 0298         MSRG                D,GR0           I------------------IE4U02980
     093C 09FF B24F 1011 0010 0100 1111 0299         JDU                 TESTE                              E4U02990
                                        0300 *-----------------------CF-----------------------------------E4U03000
                                        0301 * ESTA ROTINA CARREGA NA MP A PARTIR DO ENDERECO FF60 O CONTEUDO DE UM  E4U03010
                                        0302 *SETOR DO DISQUETE                                         E4U03020
                                        0303 *------------------------------------------------------------E4U03030
          0A00                          0304         BND                                                    E4U03040
     0CF8 0A00 A33E 1010 0011 0011 1110 0305  CF     JDS                 FETRAF                             E4U03050
0000 D    0A01 C002 1100 0000 0000 0010 0306         MID                 0,D                                E4U03060
     2008 0A02 A802 1010 1000 0000 0010 0307         JDS                 SDD                                E4U03070
0A00      0A03 0A0D 0000 1010 0000 1101 0308         DC                  FDFLAG                             E4U03080
     0A10 0A04 005B 0000 0000 0101 1011 0309         JRU                 FDWORK                             E4U03090
                                        0310 *-----------------------CF-----------------------------------E4U03100
                                        0311 * ESTA ROTINA GRAVA EM UM SETOR DO DISQUETE O CONTEUDO DA MEMORIA PRINCE4U03110
                                        0312 *A PARTIR DO ENDERECO FF60                                 E4U03120
                                        0313 *------------------------------------------------------------E4U03130
          0A08                          0314         BND                                                    E4U03140
     0CF8 0A08 A33E 1010 0011 0011 1110 0315  DF     JDS                 FETRAF                             E4U03150
0001 D    0A09 C006 1100 0000 0000 0110 0316         MID                 1,D                                E4U03160
     2008 0A0A A802 1010 1000 0000 0010 0317         JDS                 SDD                                E4U03170
0A0D      0A0B 0A0D 0000 1010 0000 1101 0318         DC                  FDFLAG                             E4U03180
     0A10 0A0C 0013 0000 0000 0001 1011 0319         JRU                 FDWORK                             E4U03190
0A0D                                    0320  FDFLAG EQU                 *             AREA BOOLEANA DE CF E CD E4U03200
          0A0D                          0321         DA                  1                                  E4U03210
          0A10                          0322         BND                                                    E4U03220
     2014 0A10 A805 1010 1000 0000 0101 0323  FDWORK JDS                 LDD                                E4U03230
0DC4      0A11 0DC4 0000 1101 1100 0100 0324         DC                  TRIL1                              E4U03240
D    GR3  0A12 21C3 0010 0001 1100 0011 0325         MSRG                D,GR3                              E4U03250
     2014 0A13 A805 1010 1000 0000 0101 0326         JDS                 LDD                                E4U03260
0D05      0A14 0D05 0000 1101 0000 0101 0327         DC                  SET2                               E4U03270
D    GR0  0A15 21C0 0010 0001 1100 0000 0328         MSRG                D,GR0                              E4U03280
0001 D    0A16 C006 1100 0000 0000 0110 0329         MID                 1,D                                E4U03290
D    GR1  0A17 21C1 0010 0001 1100 0001 0330         MSRG                D,GR1                              E4U03300
     2014 0A18 A805 1010 1000 0000 0101 0331         JDS                 LDD                                E4U03310
08C6      0A19 08C6 0000 1000 1100 0110 0332         DC                  ENDWR                              E4U03320
D    GR6  0A1A 21C6 0010 0001 1100 0110 0333         MSRG                D,GR6                              E4U03330
0080 B    0A1B C201 1100 0010 0000 0001 0334         MID                 128,B                              E4U03340
GR6  D    0A1C 2303 0010 0011 0000 0011 0335         MSGR                GR6,D                              E4U03350
     B    0A1D 1232 0001 0010 0011 0010 0336         MAR    +D R-C       B                                  E4U03360
GR0  D    0A1E 2003 0010 0000 0000 0011 0337         MSGR                GR0,D                              E4U03370
D    GR4  0A1F 21C4 0010 0001 1100 0100 0338         MSRG                D,GR4                              E4U03380
0001 D    0A20 C006 1100 0000 0000 0110 0339         MID                 1,D                                E4U03390
D    GR5  0A21 21C5 0010 0001 1100 0101 0340         MSRG                D,GR5                              E4U03400
     2014 0A22 A805 1010 1000 0000 0101 0341         JDS                 LDD                                E4U03410
0A0D      0A23 0A0D 0000 1010 0000 1101 0342         DC                  FDFLAG                             E4U03420
0001 B    0A24 C005 1100 0000 0000 0101 0343         MID                 1,B                                E4U03430
     B    0A25 1122 0001 0001 0010 0010 0344         MAF    EOP          B                                  E4U03440
S    0A29 0A26 9072 1001 0000 0111 0010 0345         TST          70     S,*+3                              E4U03450
     2870 0A27 AA1C 1010 1010 0001 1100 0346         JDS                 FDR0                               E4U03460
     0A2A 0A28 000B 0000 0000 0000 1011 0347         JRU                 *+2                                E4U03470
     2878 0A29 AA1E 1010 1010 0001 1110 0348         JDS                 FDWT                               E4U03480
```

```
                                                    C  .  1  .  2  .  3  .  4  .  5  .  6  .  7  .  8
ORIG DEST LOCN INST CONFIGURACAO        BITS NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                      ID/SEQ
     2014 0A2A A805 1010 1000 0000 0101 0349          JDS               LDO             I---------------------------EMU03490
0958      0A2B 0958 0000 1001 0101 1011 0350          DC                SLVGRO          IDEVOLVE GRO               IEMU03500
D    GRO  0A2C 2100 0010 0001 1100 0000 0351          MSRG              D,GRO           I---------------------------IEMU03510
     093C 0A2D B24F 1011 0010 0100 1111 0352          JDU               TESTE                                       EMU03520
                                        0353 *-------------------BUSCA INSTRUCAO ANTECIPADA----PREFETCH--------------EMU03530
          0A30                          0354          BND                                                           EMU03540
000F B    0A30 C030 1100 0000 0011 1101 0355  PREFET MID               X'00F',B                                    EMU03550
     0A58 0A31 A296 1010 0010 1001 0110 0356          JDS               LEREG                                       EMU03560
D    A    0A32 3184 0011 0001 1000 0100 0357          MSRR              D,A             A<--D  A<--PC               EMU03570
     0A40 0A33 A290 1010 0010 1001 0000 0358          JDS               AJENDR          AJUSTA ENDERECO             EMU03580
          0A34 0816 0000 1000 0001 0110 0359          RSET              14                                          EMU03590
B    B    0A35 3502 0011 0101 0000 0010 0360          MSRR R            B,B             D<--MEM(A)                  EMU03600
B    B    0A36 3D02 0011 1101 0000 0010 0361          MSRR B            B,B             ESPERA TERMINAR             EMU03610
D    GRO  0A37 21C0 0010 0001 1100 0000 0362          MSRG              D,GRO           GRO<--INST. ANTECIPADA      EMU03620
                                        0363 *---------------------------PC<--PC+1-------------------------------------EMU03630
000F B    0A38 C030 1100 0000 0011 1101 0364          MID               X'00F',B        B<---'000F'                 EMU03640
     0A58 0A39 A296 1010 0010 1001 0110 0365          JDS               LEREG           D<---PC                     EMU03650
D    D    0A3A 11E3 0001 0001 1110 0011 0366          MURR     +1       D,D             D<---D+1                    EMU03660
000F B    0A3B C030 1100 0000 0011 1101 0367          MID               X'00F',B        B<--'000F'                  EMU03670
     LM1  0A3C 000D 0000 0000 0000 1101 0368          MOLM              1               LM(1)<--D PC ATUALIZADO     EMU03680
     0A50 0A3D A294 1010 0010 1001 0100 0369          JDS               GRAVRG          MEM(PC)<--CONTEUDO          EMU03690
STK  MPC  0A3E 3307 0011 0011 0000 0111 0370          MSRR              STK,MPC         RETORNA                     EMU03700
                                        0371 * O ENDERECO DEVE SER COLOCADO EM A E SERA DEVOLVIDO REAJUSTADO EM A****EMU03710
          0A40                          0372          BND                                                           EMU03720
A    B    0A40 3222 0011 0010 0010 0010 0373  AJENDR MSRR     SL       A,B             B<--A DESL. ESQUERDA(1)     EMU03730
0800 D    0A41 E002 1110 0000 0000 0010 0374          MID               X'800',D        D<--'800'(2KB)              EMU03740
     A    0A42 1224 0001 0010 0010 0100 0375          MAR      +0       A               SOMA 2KB NO ENDERECO        EMU03750
STK  MPC  0A43 3307 0011 0011 0000 0111 0376          MSRR              STK,MPC         RETORNA                     EMU03760
                                        0377 *---------------------------BUSCA INSTRUCAO-------FETCH-----------------EMU03770
                                        0378 *                                                                      EMU03780
          0A44                          0379          BND                                                           EMU03790
GRO  B    0A44 2002 0010 0000 0000 0010 0380  FETCH  MSGR              GRO,B           B<--GRO                     EMU03800
B    GR1  0A45 2141 0010 0001 0100 0001 0381          MSRG              B,GR1           GR1<--B  GR1<--INST.A EXEC  EMU03810
     0A30 0A46 A20C 1010 0010 1000 1100 0382          JDS               PREFET          BUSCA INST. ANTECIPADA      EMU03820
GR1  B    0A47 2082 0010 0000 1000 0010 0383          MSGR              GR1,B           B<---GR1                    EMU03830
B    0A4C 0A48 9103 1001 0001 0000 0011 0384          TST          00   B,YYY           SE NAO IMEDIATO IR P/DECOD  EMU03840
GRO  B    0A49 2002 0010 0000 0000 0010 0385          MSGR              GRO,B           B<--INST.ANTECIP.(IMEDIAT)  EMU03850
B    GR2  0A4A 2142 0010 0001 0100 0010 0386          MSRG              B,GR2           GR2<--IMEDIATO(GRO)         EMU03860
     0A30 0A4B A20C 1010 0010 1000 1100 0387          JDS               PREFET          BUSCA INST. ANTECIPADA      EMU03870
     0A60 0A4C B293 1011 0010 1001 1000 0388  YYY    JDU               MONINS                                      EMU03880
                                        0389 *-----------------------SUBROTINA GRAVRG------------------------------*  EMU03890
                                        0390 *PARA USAR A SUBROTINA, O REGISTRADOR DESEJADO DEVE SER COLOCADO EM B   EMU03900
                                        0391 *E O SEU CONTEUDO EM LM(1)*********************************************** EMU03910
          0A50                          0392          BND                                                           EMU03920
LM0       0A50 0004 0000 0000 0000 0100 0393  GRAVRG MILM              0               D<--LM(0) D<--'FFE0'        EMU03930
B    B    0A51 3122 0011 0001 0010 0010 0394          MSRR     SL       B,B             AJUSTA REGISTRADOR          EMU03940
     A    0A52 1224 0001 0010 0010 0100 0395          MAR      +0       A               A<==D+0                     EMU03950
LM1       0A53 000C 0000 0000 0000 1100 0396          MILM              1               D<---LM(1)                  EMU03960
          0A54 0816 0000 1000 0001 0110 0397          RSET              14                                          EMU03970
B    B    0A55 3902 0011 1001 0000 0010 0398          MSRR W            B,B             MEM(REG)<--CONTEUDO         EMU03980
B    B    0A56 3D02 0011 1101 0000 0010 0399          MSRR B            B,B             ESPERA FIM DA GRAVACAO      EMU03990
STK  MPC  0A57 3307 0011 0011 0000 0111 0400          MSRR              STK,MPC                                     EMU04000
                                        0401 *-----------------------SUBROTINA LEREG---------------------------------EMU04010
                                        0402 *PARA USAR A SUBROTINA, O REGISTRADOR DESEJADO DEVE SER COLOCADO EM B.  EMU04020
                                        0403 *E O SEU CONTEUDO SERA DEVOLVIDO EM D *********************************EMU04030
          0A58                          0404          BND                                                           EMU04040
LM0       0A58 0004 0000 0000 0000 0100 0405  LEREG  MILM              0               D<--LM(0) D<--'FFE0'        EMU04050
B    B    0A59 3122 0011 0001 0010 0010 0406          MSRR     SL       B,B             AJUSTA REGISTRADOR          EMU04060
```

```
                                                         C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCN INST CONFIGURACAO        BITS NINS ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO                     ID/SEQ
     A    0A5A 1224 0001 0010 0010 0100 0407            MAR    +0              A               A<---+0                        EMU04070
          0A5B 0816 0000 1000 0001 0110 0408            RSET                   14                                             EMU04080
B    B    0A5C 3502 0011 0101 0000 0010 0409            MSRR 8                 B,B             D<==REGIST.                    EMU04090
B    B    0A5D 3D02 0011 1101 0000 0010 0410            MSRR 6                 B,B             D<==REGIST.                    EMU04100
STK  MPC  0A5E 3307 0011 0011 0000 0111 0411            MSRR                   STK,MPC                    RETORNA             EMU04110
                                        0412 *--------------D E C O D I F I C A C A O   D A S   I N S T R U C O E S----EMU04120
          0A60                          0413            BND                                                                   EMU04130
     0CEC 0A60 A333 1010 0011 0011 1011 0414 MONINS JOS                    ROTMON                                         EMU04140
0E17      0A61 0E17 0000 1110 0001 0111 0415            DC                     CTINST                                         EMU04150
GR1  B    0A62 2082 0010 0000 1000 0010 0416            MSGR                   GR1,B                                          EMU04160
B    0A67 0A63 9103 1001 0001 0000 0011 0417            TST           00       B,CNTNIM                                       EMU04170
     0CEC 0A64 A333 1010 0011 0011 1011 0418            JOS                    ROTMON                                         EMU04180
0E18      0A65 0E18 0000 1110 0001 1000 0419            DC                     CTIM                                           EMU04190
          0A66 0000 0000 0000 0000 0000 0420            JRU                    DECOD                                          EMU04200

     0CEC 0A67 A333 1010 0011 0011 1011 0421 CNTNIM JOS                    ROTMON                                         EMU04210
0E19      0A68 0E19 0000 1110 0001 1001 0422            DC                     CTNIM                                          EMU04220
GR1  B    0A69 20A2 0010 0000 1010 0010 0423 DECOD  MSGR   SL              GR1,B           B<--GR1 DESLOCADO(1)           EMU04230
B    B    0A6A 6172 0110 0001 0111 0010 0424            MORM   2  XSL          B,B             B<--CODIGO DA INSTRUCAO        EMU04240
0A6E D    0A6B E9BA 1110 1001 1011 1010 0425            MIO                    TABDEV,D        D<--TABDEV(ENDERECO)           EMU04250
     D    0A6C 1223 0001 0010 0010 0011 0426            MAR    +0              D               D<--TABDEV+CODIGO-INSTRUC.     EMU04260
D    MPC  0A6D 3187 0011 0001 1000 0111 0427            MSRR                   D,MPC           MPC<--ENDERECO DA ROTINA       EMU04270
     0A80 0A6E B2A0 1011 0010 1010 0000 0428 TABDEV JOU                    OP                                             EMU04280
     0AA4 0A6F B2E9 1011 0010 1110 1001 0429            JOU                    OPK                                            EMU04290
     0BCC 0A70 B2F3 1011 0010 1111 0011 0430            JOU                    IE                                             EMU04300
     0C00 0A71 B330 1011 0011 0000 0000 0431            JOU                    SEQ                                            EMU04310
     0C18 0A72 B306 1011 0011 0000 0110 0432            JOU                    SGT                                            EMU04320
     0C20 0A73 B308 1011 0011 0000 1000 0433            JOU                    SGE                                            EMU04330
     0C34 0A74 B30D 1011 0011 0000 1101 0434            JOU                    SF                                             EMU04340
     0C40 0A75 B310 1011 0011 0001 0000 0435            JOU                    RC                                             EMU04350
     0C90 0A76 B324 1011 0011 0010 0100 0436            JOU                    STORE                                          EMU04360
          0A77 0000 0000 0000 0000 0000 0437            NOP                                                                   EMU04370
     0CA4 0A78 B329 1011 0011 0010 1001 0438            JOU                    PUSH                                           EMU04380
          0A79 0000 0000 0000 0000 0000 0439            NOP                                                                   EMU04390
     0CC0 0A7A B330 1011 0011 0011 0000 0440            JOU                    LOAD                                           EMU04400
          0A7B 0000 0000 0000 0000 0000 0441            NOP                                                                   EMU04410
     0CD0 0A7C B334 1011 0011 0011 0100 0442            JOU                    POP                                            EMU04420
STK  MPC  0A7D 3307 0011 0011 0000 0111 0443            MSRR                   STK,MPC         VOLTA P/ GERENCIADOR           EMU04430
                                        0444 *--------------R O T I N A S   D A S   I N S T R U C O E S--------------EMU04440
                                        0445 *--------------------------INSTRUCAO OP-----------------------------------EMU04450
          0A80                          0446            BND                                                                   EMU04460
GR1  B    0A80 2082 0010 0000 1000 0010 0447 OP     MSGR                   GR1,B           D<--INSTRUCAO                  EMU04470
     0CEC 0A81 A333 1010 0011 0011 1011 0448            JOS                    ROTMON                                         EMU04480
0E1A      0A82 0E1A 0000 1110 0001 1010 0449            DC                     CTOP                                           EMU04490
B    0A85 0A83 9101 1001 0001 0000 0001 0450            TST           00       B,*+2                                          EMU04500
     0A94 0A84 B2A5 1011 0010 1010 0101 0451            JOU                    OPIM            SE B(0)=1 IR P/OPIM            EMU04510
GR1  B    0A85 40FA 0100 0000 1111 1010 0452            MOGR   4  XSR          GR1,B           B<--RD                         EMU04520
     0A58 0A86 A296 1010 0010 1001 0110 0453            JOS                    LEREG                                          EMU04530
D    GR4  0A87 21C4 0010 0001 1100 0100 0454            MSRG                   D,GR4           GR4<--(RD)                     EMU04540
GR1  B    0A88 40E2 0100 0000 1110 0010 0455            MOGR   4               GR1,B           B<--RS                         EMU04550
     0A58 0A89 A296 1010 0010 1001 0110 0456            JOS                    LEREG                                          EMU04560
D    GR3  0A8A 21C3 0010 0001 1100 0011 0457            MSRG                   D,GR3           GR3<--RS                       EMU04570
     0A98 0A8B A2A6 1010 0010 1010 0110 0458            JOS                    OPER                                           EMU04580
GR3  D    0A8C 2133 0010 0001 0011 0011 0459            MSGR                   GR3,D                                          EMU04590
GR1  B    0A8D 40FA 0100 0000 1111 1010 0460            MOGR   4  XSR          GR1,B           B<--RD                         EMU04600
     LM1  0A8E 0000 0000 0000 0000 1101 0461            MCLM                   1                                              EMU04610
```

```
                                                  C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCN INST CONFIGURACAO        NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                     ID/SEQ
     0A50 0A6F A294 1010 0010 1001 0100 0462         JDS                GRAVRG                                         EMU04620
          0A70 0000 0000 0000 0000 0000 0463         NOP                                 "JDS ARITH"                   EMU04630
     093C 0A91 B24F 1011 0010 0100 1111 0464         JDU                TESTE                                          EMU04640
          0A94                          0465         END                                                               EMU04650
GR2  D    0A94 2103 0010 0001 0000 0011 0466  OPIM   MSGR               GR2,D            D<--IMEDIATO                  EMU04660
D    GR4  0A95 21C4 0010 0001 1100 0100 0467         MSRG               D,GR4            GR4<--IMEDIATO                EMU04670
     0A85 0A96 0773 0000 0111 0111 0011 0468         JRU                OP+5                                           EMU04680
                                        0469  *-----------------------SUBROTINA OPER-----------------------------------EMU04690
          0A98                          0470         END                                                               EMU04700
GR1  B    0A98 40EA 0100 0000 1110 1010 0471  OPER   MOGR   4  X        GR1,B            B<--CODIGO OPERACAO+BIT12     EMU04710
0007 D    0A99 C01E 1100 0000 0001 1110 0472         MID                X'007',D         D<--'7' P/ZERAR BIT 12        EMU04720
     B    0A9A 10A2 0001 0000 1010 0010 0473         MAR    AND         B                B<--CODIGO DA OPERACAO        EMU04730
0A9E D    0A9B EA7A 1110 1010 0111 1010 0474         MID                TABOP,D          D<--TABOP(ENDERECO)           EMU04740
     B    0A9C 1222 0001 0010 0010 0010 0475         MAR    +D          B                B<--TABOP+CODIGO OPER         EMU04750
B    MPC  0A9D 3107 0011 0001 0000 0111 0476         MSRA               B,MPC            MPC<--ENDERECO DA ROTINA      EMU04760
     0AA8 0A9E B2AA 1011 0010 1010 1010 0477  TABOP  JDU                ADD                                            EMU04770
     0AB8 0A9F B2AE 1011 0010 1010 1110 0478         JDU                ADC                                            EMU04780
     0AC4 0AA0 B2B1 1011 0010 1011 0001 0479         JDU                SUB                                            EMU04790
     0ACC 0AA1 B2B3 1011 0010 1011 0011 0480         JDU                SBC                                            EMU04800
     0AD8 0AA2 B2B6 1011 0010 1011 0110 0481         JDU                ADDI                                           EMU04810
     0AF0 0AA3 B2BC 1011 0010 1011 1100 0482         JDU                [illegible]                                    EMU04820
     0AF8 0AA4 B2BE 1011 0010 1011 1110 0483         JDU                OR                                             EMU04830
     0B00 0AA5 B2C0 1011 0010 1100 0000 0484         JDU                XOR                                            EMU04840
                                        0485  *--------------R O T I N A S   D A S   O P E R A C O E S (ULA)-----------EMU04850
                                        0486  *---------------------------OPERACAO ADD---------------------------------EMU04860
          0AA8                          0487         END                                                               EMU04870
     0CEC 0AA8 A33B 1010 0011 0011 1011 0488  ADD    JDS                ROTMON                                         EMU04880
0E1B      0AA9 0E1B 0000 1110 0001 1011 0489         DC                 CTADD                                          EMU04890
GR3  B    0AAA 2182 0010 0001 1000 0010 0490         MSGR               GR3,B            B<--GR3                       EMU04900
GR4  D    0AAB 2203 0010 0010 0000 0011 0491         MSGR               GR4,D            D<--GR4                *ADD+1*EMU04910
     GR3  0AAC 120B 0001 0010 0000 1011 0492         MAG    +D D-3      GR3              GR3<--B+D  S(3)<--VAI-UM      EMU04920
B    B    0AAD 3122 0011 0001 0010 0010 0493         MSRR   DL          B,B                                     *ADD+3*EMU04930
B    B    0AAE 312A 0011 0001 0010 1010 0494         MSRR   DR          B,B              ZERA BIT 15 DE B              EMU04940
D    D    0AAF 3143 0011 0001 0100 0011 0495         MSRR   DL          D,D                                            EMU04950
D    D    0AB0 31AB 0011 0001 1010 1011 0496         MSRR   DR          D,D              ZERA BIT 15 DE D              EMU04960
     D    0AB1 1223 0001 0010 0010 0011 0497         MAR    +D          D                D<--B+D                       EMU04970
D    0AB5 0AB2 95B2 1001 0101 1011 0010 0498         TST             11 D,C15            SE D(3)=1 IR P/'C15'          EMU04980
0000 B    0AB3 C001 1100 0000 0000 0001 0499         MID                X'000',B         B<--'0000'                    EMU04990
     0AB6 0AB4 000B 0000 0000 0000 1011 0500         JRU                C15+1            IR P/'C15+1'                  EMU05000
0001 B    0AB5 C005 1100 0000 0000 0101 0501  C15    MID                X'001',B         B<--'0001'                    EMU05010
B    GR4  0AB6 2144 0010 0001 0100 0100 0502         MSRG               B,GR4            GR4<--B                *C15+1*EMU05020
STK  MPC  0AB7 3307 0011 0011 0000 0111 0503         MSRA               STK,MPC          RETORNA                       EMU05030
                                        0504  *---------------------OPERACAO ADD COM CARRY-----------------------------EMU05040
          0AB8                          0505         END                                                               EMU05050
0000 B    0AB8 C035 1100 0000 0011 0101 0506  ADC    MID                X'000',B                                       EMU05060
     0CEC 0AB9 A33B 1010 0011 0011 1011 0507         JDS                ROTMON                                         EMU05070
0E1C      0ABA 0E1C 0000 1110 0001 1100 0508         DC                 CTADDC                                         EMU05080
     0A53 0ABB A296 1010 0010 1001 0110 0509         JDS                LEREG                                          EMU05090
D    0ABF 0ABC 95A2 1001 0101 1010 0010 0510         TST             21 D,VALCAR         SE ST(13)=1 IR P/'VALCAR'     EMU05100
0000 B    0ABD C001 1100 0000 0000 0001 0511         MID                X'000',B         B<--CARRY(0)                  EMU05110
     0AC0 0ABE 000B 0000 0000 0000 1011 0512         JRU                VALCAR+1         IR P/'VALCAR+1'               EMU05120
0001 B    0ABF C005 1100 0000 0000 0101 0513  VALCAR MID                X'001',B         B<--CARRY(1)                  EMU05130
GR3  D    0AC0 2183 0010 0001 1000 0011 0514         MSGR               GR3,D            D<--RS              *VALCAR+1*EMU05140
     B    0AC1 1222 0001 0010 0010 0010 0515         MAR    +D          B                B<--RS+CARRY                  EMU05150
     0AAB 0AC2 0743 0000 0111 0100 0011 0516         JRU                ADD+3            IR P/'ADD+3'                  EMU05160
                                        0517  *---------------------------OPERACAO SUB---------------------------------EMU05170
          0AC4                          0518         END                                                               EMU05180
GR3  B    0AC4 2182 0010 0001 1000 0010 0519  SUB    MSGR               GR3,B            B<--GR3(RS)                   EMU05190
```

```
                                      C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCN INST CONFIGURACAO        NIMS ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO                 ID/SEQ
     0CEC 0AC5 A33B 1010 0011 0011 1011 0520              JDS                  ROTMON                                       EMU05200
0E10      0AC6 0E10 0000 1110 0001 1101 0521              DC                   CTSUB                                        EMU05210
GR4  D    0AC7 2203 0010 0010 0000 0011 0522              MSGR                 GR4,D           D<--GR4(RD) OU IMED  *SUB+1*EMU05220
     GR3  0AC8 1303 0001 0011 0000 1011 0523              MAC     -1 0-3       GR3             GR3<--D-E, S(Z)<--VAI-UM     EMU05230
     0AA0 0AC9 0713 0000 0111 0001 1011 0524              JRU                  ADD+5           IP P/'ADD+5'                 EMU05240
                                        0525 ------------------OPERACAO SUB COM CARRY----------------------------------------EMU05250
          0ACC                          0526              BND                                                               EMU05260
0000 B    0ACC C035 1100 0000 0011 0101 0527 SBC          MID                  X'000',3                                     EMU05270
     0CEC 0ACD A33B 1010 0011 0011 1011 0528              JDS                  ROTMON                                       EMU05280
0E1E      0ACE 0E1E 0000 1110 0001 1110 0529              DC                   CTSBC                                        EMU05290
     0A53 0ACF A296 1010 0010 1001 0110 0530              JDS                  LEREG                                        EMU05300
D    0A03 0AD0 95A2 1001 0101 1010 0010 0531              TST           21     D,VCARRY        SE ST(13)=1 IP P/'VCARRY'    EMU05310
0000 B    0AD1 C001 1100 0000 0000 0001 0532              MID                  X'000',B        B<--CARRY(0)                 EMU05320
     0AD4 0AD2 0008 0000 0000 0000 1011 0533              JRU                  VCARRY+1        IP P/'VCARRY+1'              EMU05330
0001 B    0AD3 C005 1100 0000 0000 0101 0534 VCARRY       MID                  X'001',B        B<--CARRY(1)                 EMU05340
GR3  D    0AD4 2183 0010 0001 1000 0011 0535              MSGR                 GR3,D           D<--RS         *VCARRY+1*EMU05350
     B    0AD5 1322 0001 0011 0010 0010 0536              MAR     -1           D               B<--RS-CARRY                 EMU05360
     0AC7 0AD6 0783 0000 0111 1000 0011 0537              JRU                  SUB+3           IP P/'SUB+3'                 EMU05370
                                        0538 ------------------OPERACAO ADD COM INDICE----------------------------------------EMU05380
          0AD8                          0539              BND                                                               EMU05390
     0CEC 0AD8 A33B 1010 0011 0011 1011 0540 ADDI         JDS                  ROTMON                                       EMU05400
0E1F      0AD9 0E1F 0000 1110 0001 1111 0541              DC                   CTADI                                        EMU05410
GR3  A    0ADA 21AC 0010 0001 1010 1100 0542              MSGR    SR           GR3,A           A<--S(DESL) S(4)<--BIT15     EMU05420
S    0AE6 0ADB 944A 1001 0100 0100 1010 0543              TST           41     S,INDUM         IP P/'INDUM'                 EMU05430
0000 B    0ADC C035 1100 0000 0011 0101 0544              MID                  X'000',3                                     EMU05440
     0A53 0ADD A296 1010 0010 1001 0110 0545              JDS                  LEREG                                        EMU05450
     LM2  0ADE 0015 0000 0000 0001 0101 0546              MOLM                 2                                            EMU05460
00FE D    0ADF C3FA 1100 0011 1111 1010 0547              MID                  X'0FE',D        D<--'0000000011111110'       EMU05470
00FF B    0AE0 C3FD 1100 0011 1111 1101 0548              MID                  X'0FF',B        B<--'0000000011111111'       EMU05480
D    D    0AE1 313B 0011 0001 1000 1011 0549              MSRR    X            D,D             D<--'1111111011111111'       EMU05490
     B    0AE2 1222 0001 0010 0010 0010 0550              MAR     +0           B                                            EMU05500
LM2       0AE3 0014 0000 0000 0001 0100 0551              MILM                 2               D<--LM(2)                    EMU05510
     D    0AE4 10A3 0001 0000 1010 0011 0552              MAR     AND          D               D<--ST(COM IND-ZERO)ST8<-0   EMU05520
     0AE4 0AE5 0F23 0000 0000 0010 0011 0553              JRU                  INDUM+4                                      EMU05530
0000 B    0AE6 C035 1100 0000 0011 0101 0554 INDUM        MID                  X'000',3        IP P/'INDUM'                 EMU05540
     0A56 0AE7 A296 1010 0010 1001 0110 0555              JDS                  LEREG                                        EMU05550
0100 B    0AE8 C401 1100 0100 0000 0001 0556              MID                  X'100',3                                     EMU05560
     D    0AE9 1143 0001 0001 0100 0011 0557              MAR     OR           D                                            EMU05570
0000 B    0AEA C035 1100 0000 0011 0101 0558              MID                  X'000',3                   *INDUM+4*EMU05580
     LM1  0AEB 0030 0000 0000 0000 1101 0559              MOLM                 1                                            EMU05590
     0A5C 0AEC A294 1010 0010 1001 0100 0560              JDS                  GRAVRG                                       EMU05600
GR3  A    0AED 21AC 0010 0001 1010 1100 0561              MSGR    SR           GR3,A                                        EMU05610
A    GR3  0AEE 2243 0010 0010 0100 0011 0562              MSRG                 A,GR3                                        EMU05620
     0AAA 0AEF 0503 0000 0101 1101 0011 0563              JRU                  ADD+2                                        EMU05630
                                        0564 ------------------OPERACAO AND-----------------------------------------------------EMU05640
          0AF0                          0565              BND                                                               EMU05650
GR3  B    0AF0 2182 0010 0001 1000 0010 0566 AND          MSGR                 GR3,B           B<--RS                       EMU05660
     0CEC 0AF1 A33B 1010 0011 0011 1011 0567              JDS                  ROTMON                                       EMU05670
0E20      0AF2 0E20 0000 1110 0010 0000 0568              DC                   CTAND                                        EMU05680
GR4  D    0AF3 2203 0010 0010 0000 0011 0569              MSGR                 GR4,D           D<---RD(OU IMEDIATO)         EMU05690
     GR3  0AF4 1083 0001 0000 1000 0011 0570              MAC     AND          GR3             GR3<--RS AND RD              EMU05700
0000 B    0AF5 C001 1100 0000 0000 0001 0571              MID                  X'000',B        B<---'0000'                  EMU05710
B    GR4  0AF6 2144 0010 0001 0100 0100 0572              MSRG                 B,GR4           GR4<---'0000'                EMU05720
STK  MPE  0AF7 3307 0011 0011 0000 0111 0573              MSRR                 STK,MPC         RETORNA                      EMU05730
                                        0574 ------------------OPERACAO OR------------------------------------------------------EMU05740
          0AF8                          0575              BND                                                               EMU05750
GR3  B    0AF8 2182 0010 0001 1000 0010 0576 OR           MSGR                 GR3,B           B<--RS                       EMU05760
     0CEC 0AF9 A33B 1010 0011 0011 1011 0577              JDS                  ROTMON                                       EMU05770
```

206


```
                                                        C  .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST  LOCN  INST  CONFIGURACAO   BITS  NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                    ID/SEQ
0E21       0AFA  0E21  0000 1110 0010 0001  0578         DC                  CTOR                                          EMU05780
GR4  D     0AFB  2203  0010 0010 0000 0011  0579         MSGR                GR4,D          D<---RD(OU IMEDIATO)           EMU05790
     GR3   0AFC  1183  0001 0001 1000 0011  0580         MAG    OR           GR3            GR3<--RS OR RD(IMEDIATO)       EMU05800
0000 B     0AFD  C001  1100 0000 0000 0001  0581         MID                 X'000',B       B<---'0000'                    EMU05810
B    GR4   0AFE  2144  0010 0001 0100 0100  0582         MSRG                B,GR4          GR4<---'0000'                  EMU05820
STK  MPC   0AFF  3307  0011 0011 0000 0111  0583         MSRR                STK,MPC        RETORNA                        EMU05830
                                                  0584  *------------------------------OPERACAO XOR---------------------------EMU05840
           0800                                   0585         RND                                                             EMU05850
GR3  B     0800  2182  0010 0001 1000 0010  0586  XOR    MSGR                GR3,B          B<---RS                        EMU05860
     0CEC  0801  A33B  1010 0011 0011 1011  0587         JDS                 ROTMON                                        EMU05870
0E22       0802  CE22  0000 1110 0010 0010  0588         DC                  CTXOR                                         EMU05880
GR4  D     0803  2203  0010 0010 0000 0011  0589         MSGR                GR4,D          D<---RD(OU IMEDIATO)           EMU05890
     GR3   0804  1103  0001 0001 0000 0011  0590         MAG    XOR          GR3            GR3<--RS XOR RD(OU IMEDIATO)EMU05900
0000 B     0805  C001  1100 0000 0000 0001  0591         MID                 X'000',B       B<---'0000'                    EMU05910
B    GR4   0806  2144  0010 0001 0100 0100  0592         MSRG                B,GR4          GR4<---'0000'                  EMU05920
STK  MPC   0807  3307  0011 0011 0000 0111  0593         MSRR                STK,MPC        RETORNA                        EMU05930
                                                  0594  *-----------------------SUBROTINA ARITM----------------------------EMU05940
           0808                                   0595         RND                                                             EMU05950
GR3  D     0808  2183  0010 0001 1000 0011  0596  ARIT   MSGR                GR3,D          D<---RESULTADO                 EMU05960
D    0816  0809  955C  1001 0101 1000 1100  0597         TST           01    D,NEGATI       SE  ITC=1 IR P/'NEGATI'        EMU05970
0000 B     080A  C035  1100 0000 0011 0101  0598         MID                 X'000',B                                      EMU05980
     0A56  080B  A296  1010 0010 1001 0110  0599         JDS                 LEREG                                         EMU05990
     LM2   080C  0015  0000 0000 0001 0101  0600         MOLM                2                                             EMU06000
00FF B     080D  C3FD  1100 0011 1111 1101  0601         MID                 X'0FF',B       B<--'0000000011111111'         EMU06010
007F D     080E  C1FE  1100 0001 1111 1110  0602         MID                 X'07F',D       D<--'0000000001111111'         EMU06020
D    D     080F  313B  0011 0001 1000 1011  0603         MSRR   X            D,D            D<--'1111111100000000'         EMU06030
     B     0810  1222  0001 0010 0010 0010  0604         MAR    +D           B              B<--'1111111111111111'         EMU06040
LM2        0811  0014  0000 0000 0001 0100  0605         MILM                2                                             EMU06050
     D     0812  10A3  0001 0000 1010 0011  0606         MAR    AND          D              D<--ST AND B   D(0)<--0        EMU06060
0000 B     0813  C001  1100 0000 0000 0001  0607         MID                 X'000',B       B<--'0000'                     EMU06070
B    GR5   0814  2145  0010 0001 0100 0101  0608         MSRG                B,GR5          GR5<--0(NAO NEGATIVO)          EMU06080
     0829  0815  009A  0000 0000 1001 1011  0609         JMP                 NEGATI+19      IR PARA 'NEGATI+19'            EMU06090
0000 B     0816  CC35  1100 0000 0011 0101  0610  NEGATI MID                 X'000',B                                      EMU06100
     0A56  0817  A296  1010 0010 1001 0110  0611         JDS                 LEREG                                         EMU06110
     LM2   0818  0015  0000 0000 0001 0101  0612         MOLM                2                                             EMU06120
0080 B     0819  C201  1100 0010 0000 0001  0613         MID                 X'080',B       B<--'0000000010000000'         EMU06130
B    B     081A  310A  0011 0001 0000 1010  0614         MSRR   X            B,B            B<--'1000000000000000'         EMU06140
LM2        081B  0014  0000 0000 0001 0100  0615         MILM                2                                             EMU06150
     D     081C  10A3  0001 0000 1010 0011  0616         MAR    OR           D              D<--ST OR B   D(0)<--1         EMU06160
0000 B     081D  CC35  1100 0000 0011 0101  0617         MID                 X'000',B                                      EMU06170
     LM1   081E  0000  0000 0000 0000 1101  0618         MOLM                1                                             EMU06180
     0A50  081F  A294  1010 0010 1001 0100  0619         JDS                 GRAVRG                                        EMU06190
0000 B     0820  C035  1100 0000 0011 0101  0620         MID                 X'000',B                                      EMU06200
     0A56  0821  A296  1010 0010 1001 0110  0621         JDS                 LEREG                                         EMU06210
     LM2   0822  0015  0000 0000 0001 0101  0622         MOLM                2                                             EMU06220
0001 B     0823  C005  1100 0000 0000 0101  0623         MID                 X'001',B                                      EMU06230
B    GR5   0824  2145  0010 0001 0100 0101  0624         MSRG                B,GR5          GR5<---1(IS NEGATIVO)          EMU06240
00FD B     0825  C3F5  1100 0011 1111 0101  0625         MID                 X'0FD',B                                      EMU06250
B    B     0826  310A  0011 0001 0000 1010  0626         MSRR   X            B,B                                           EMU06260
00FF D     0827  C3FE  1100 0011 1111 1110  0627         MID                 X'0FF',D                                      EMU06270
     B     0828  1222  0001 0010 0010 0010  0628         MAR    +D           B              GE<---0(DESLIGA > OU =)        EMU06280
LM2        0829  0014  0000 0000 0001 0100  0629         MILM                2                                             EMU06290
     D     082A  10A3  0001 0000 1010 0011  0630         MAR    AND          D                                             EMU06300
0000 B     082B  CC35  1100 0000 0011 0101  0631         MID                 X'000',B                                      EMU06310
     LM1   082C  0000  0000 0000 0000 1101  0632         MOLM                1                                             EMU06320
     0A50  082D  A294  1010 0010 1001 0100  0633         JDS                 GRAVRG                            *NEGATI+13*EMU06330
           082E  0000  0000 0000 0000 0000  0634         NOP                                                               EMU06340
           082F  0000  0000 0000 0000 0000  0635         NOP                                                               EMU06350
```

```
                                                        C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCN INST CONFIGURACAO        MINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                 ID/SEQ
          0B30 0000 0000 0000 0000 0000 0636         NOP                                                           EMU06360
                                        0637  *TESTA OVERFLOW******************************                        EMU06370
S    0B34 0B31 9432 1001 0100 0011 0010 0638         TST         31    S,CARUM          SE S(3)=1 IR P/'CARUM'     EMU06380
000D B    0B32 C001 1100 0000 0000 0001 0639         MID               X'000',B         B<--'0000'(S(3))           EMU06390
     0B35 0B33 0003 0000 0000 0000 1011 0640         JRU               CARUM+1          IR P/ 'CARUM+1'            EMU06400
0001 B    0B34 C005 1100 0000 0000 0101 0641  CARUM  MID               X'001',B         B<---'0001'(S(3))          EMU06410
GR4  D    0B35 2203 0010 0010 0000 0011 0642         MSGR              GR4,D            D<---C15        *CARUM+1*  EMU06420
     B    0B36 1122 0001 0001 0010 0010 0643         MAR      EOR      B                B<--C15 XOR S(3)           EMU06430
B    0B42 0B37 977A 1001 0111 0111 1010 0644         TST         F1    B,OVERFL         SE B(15)=1 IR P/'OVERFL'   EMU06440
000D B    0B38 C035 1100 0000 0011 0101 0645         MID               X'000',B                                    EMU06450
     0A56 0B39 A296 1010 0010 1001 0110 0646         JDS               LEREG                                       EMU06460
     LM2  0B3A 0015 0000 0000 0001 0101 0647         MOLM              2                                           EMU06470
006F D    0B3B C2FE 1100 0010 1111 1110 0648         MID               X'06F',D         D<--'0000000001101111'     EMU06480
D    D    0B3C 313B 0011 0001 1000 1011 0649         MSRR     X        D,D              D<--'1011111100000000'     EMU06490
00FF B    0B3D C3FD 1100 0011 1111 1101 0650         MID               X'0FF',B         B<--'0000000011111111'     EMU06500
     B    0B3E 1222 0001 0010 0010 0010 0651         MAR      +0       B                B<--'1011111111111111'     EMU06510
LM2       0B3F 0014 0000 0000 0001 0100 0652         MILM              2                                           EMU06520
     D    0B40 10A3 0001 0000 1010 0011 0653         MAR      AND      D                D<--ST AND B, D(1)<--0     EMU06530
     0B42 0B41 0033 0000 0000 0011 0011 0654         JRU               OVERFL+6         IR P/'OVERFL+6'            EMU06540
000D B    0B42 C035 1100 0000 0011 0101 0655  OVERFL MID               X'000',B                                    EMU06550
     0A53 0B43 A296 1010 0010 1001 0110 0656         JDS               LEREG                                       EMU06560
     LM2  0B44 0015 0000 0000 0001 0101 0657         MOLM              2                                           EMU06570
0040 B    0B45 C101 1100 0001 0000 0001 0658         MID               X'040',B         B<---'0000000001000000'    EMU06580
B    B    0B46 310A 0011 0001 0000 1010 0659         MSRR     X        B,B              B<--'0100000000000000'     EMU06590
LM2       0B47 0014 0000 0000 0001 0100 0660         MILM              2                                           EMU06600
     D    0B48 11A3 0001 0001 1010 0011 0661         MAR      OR       D                D<--ST OR B, D(1)<--1      EMU06610
000D B    0B49 C035 1100 0000 0011 0101 0662         MID               X'000',B                                    EMU06620
     LM1  0B4A 000D 0000 0000 0000 1101 0663         MOLM              1                                           EMU06630
     0A50 0B4B A294 1010 0010 1001 0100 0664         JDS               GRAVRG                                      EMU06640
                                        0665  *TESTA CARRY*************************************                    EMU06650
S    0B56 0B4C 9439 1001 0100 0011 1001 0666         TST         31    S,CARRY1         SE S(3)=1 IR P/'CARRY1'    EMU06660
000D B    0B4D C035 1100 0000 0011 0101 0667         MID               X'000',B                                    EMU06670
     0A53 0B4E A296 1010 0010 1001 0110 0668         JDS               LEREG                                       EMU06680
     LM2  0B4F 0015 0000 0000 0001 0101 0669         MOLM              2                                           EMU06690
00DF B    0B50 C37D 1100 0011 0111 1101 0670         MID               X'0DF',B         B<---'0000000011011111'    EMU06700
B    B    0B51 310A 0011 0001 0000 1010 0671         MSRR     X        B,B              B<--'1101111100000000'     EMU06710
00FF D    0B52 C3FE 1100 0011 1111 1110 0672         MID               X'0FF',D         D<---'0000000011111111'    EMU06720
     B    0B53 1222 0001 0010 0010 0010 0673         MAR      +0       B                B<---'1101111111111111'    EMU06730
LM2       0B54 0014 0000 0000 0001 0100 0674         MILM              2                D<--LM(13), D<--ST         EMU06740
     D    0B55 10A3 0001 0000 1010 0011 0675         MAR      AND      D                D<--ST AND B, D(2)<--0 (C) EMU06750
000D B    0B56 C035 1100 0000 0011 0101 0676  CARRY1 MID               X'000',B                                    EMU06760
     0A53 0B57 A296 1010 0010 1001 0110 0677         JDS               LEREG                                       EMU06770
     LM2  0B58 0015 0000 0000 0001 0101 0678         MOLM              2                                           EMU06780
0020 B    0B59 C081 1100 0000 1000 0001 0679         MID               X'020',B         B<--'0000000000100000'     EMU06790
B    B    0B5A 310A 0011 0001 0000 1010 0680         MSRR     X        B,B              B<--'0010000000000000'     EMU06800
LM2       0B5B 0014 0000 0000 0001 0100 0681         MILM              2                                           EMU06810
     D    0B5C 11A3 0001 0001 1010 0011 0682         MAR      OR       D                D<--ST OR B, D(2)<--1      EMU06820
000D B    0B5D C035 1100 0000 0011 0101 0683         MID               X'000',B                                    EMU06830
     LM1  0B5E 000D 0000 0000 0000 1101 0684         MOLM              1                                           EMU06840
     0A50 0B5F A294 1010 0010 1001 0100 0685         JDS               GRAVRG                                      EMU06850
     LM13 0B60 006D 0000 0000 0110 1101 0686         MOLM              13               LM(13)<--D(ST)  *CARRY1+4* EMU06860
                                        0687  *TESTA SE IGUAL A ZERO***********************************            EMU06870
GR7  B    0B61 2332 0010 0011 1000 0010 0688         MSGR              GR7,B            B<--GR7                    EMU06880
B    0B70 0B62 937D 1001 0011 0111 1101 0689         TST         F0    B,EZERO          SE B(15)=1 IR P/'EZERO'    EMU06890
0000 B    0B63 C001 1100 0000 0000 0001 0690         MID               X'000',B         B<---'0000'                EMU06900
B    GR6  0B64 2146 0010 0001 0100 0110 0691         MSRG              B,GR6            GR6<--0 (NOT IS ZERO)      EMU06910
000D B    0B65 C035 1100 0000 0011 0101 0692         MID               X'000',B                                    EMU06920
     0A53 0B66 A296 1010 0010 1001 0110 0693         JDS               LEREG                                       EMU06930
```

```
                                                               C    .    1    .    2    .    3    .    4    .    5    .    6    .    7    .    8
ORIG DEST LOCT INST CONFIGURACAO        NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO   COMENTARIO                              ID/SEQ
     LM2  0867 0015 0000 0000 0001 0101 0694         MOLM              2                                                          EMU06940
00F7 B    0868 C3D0 1100 0011 1101 0101 0695         MID               X'0F7',B                                                   EMU06950
B    B    0869 310A 0011 0001 0000 1010 0696         MSRR   X          B,B                                                        EMU06960
00FF D    086A C3FE 1100 0011 1111 1110 0697         MID               X'0FF',D                                                   EMU06970
          086B 0000 0000 0000 0000 0000 0698         NOP                                                                          EMU06980
     B    086C 1222 0001 0010 0010 0010 0699         MAR    +0         B                                                          EMU06990
LM2       086D 0014 0000 0000 0001 0100 0700         MILM              2                 B<---'1111011111111111'                  EMU07000
     D    086E 10A3 0001 0000 1010 0011 0701         MAR    AND        D                                                          EMU07010
     0884 086F 00A3 0000 0000 1010 0011 0702         JRU               XXX               D<---ST AND B, P(4)<--0(EQ)              EMU07020
0000 B    0870 C035 1100 0000 0011 0101 0703  EZERO  MID               X'000',B                                                   EMU07030
     0A55 0871 A296 1010 0010 1001 0110 0704         JDS               LEREG                                                      EMU07040
     LM2  0872 0015 0000 0000 0001 0101 0705         MOLM              2                                                          EMU07050
0200 B    0873 C801 1100 1000 0000 0001 0706         MID               X'200',B          B<--'0000001000000000'                   EMU07060
LM2       0874 0014 0000 0000 0001 0100 0707         MILM              2                                                          EMU07070
     D    0875 11A3 0001 0001 1010 0011 0708         MAR    OR         D                                                          EMU07080
0000 B    0876 C035 1100 0000 0011 0101 0709         MID               X'000',B          D<--ST OR B, P(6)<--1 (GE)               EMU07090
     LM1  0877 0000 0000 0000 0000 1101 0710         MOLM              1                                                          EMU07100
     0A50 0878 A294 1010 0010 1001 0100 0711         JDS               GRAVRG                                                     EMU07110
0001 B    0879 C005 1100 0000 0000 0101 0712         MID               X'001',B          B<---'0001'                              EMU07120
B    GR6  087A 2146 0010 0001 0100 0110 0713         MSRG              B,GR6             GR6<---1 (IF ZERO)                       EMU07130
0000 B    087B C035 1100 0000 0011 0101 0714         MID               X'000',B                                                   EMU07140
     0A56 087C A296 1010 0010 1001 0110 0715         JDS               LEREG                                                      EMU07150
     LM2  087D 0015 0000 0000 0001 0101 0716         MOLM              2                                                          EMU07160
0060 B    087E C201 1100 0010 0000 0001 0717         MID               X'060',B          B<---'[illegible]'                       EMU07170
LM2       087F 0014 0000 0000 0001 0100 0718         MILM              2                                                          EMU07180
     D    0880 11A3 0001 0001 1010 0011 0719         MAR    OR         D                                                          EMU07190
0000 B    0881 C035 1100 0000 0011 0101 0720         MID               X'000',B          D<--ST OR B, P(4)<--1  (EQ)              EMU07200
     LM1  0882 0000 0000 0000 0000 1101 0721         MOLM              1                                                          EMU07210
     0A50 0883 A294 1010 0010 1001 0100 0722         JDS               GRAVRG                                        (EQ)         EMU07220
                                        0723  * TESTA SE MAIOR QUE ZERO*******************************************************    EMU07230
GR5  B    0884 2232 0010 0010 1000 0011 0724  XXX    MSGR              GR5,B             D<--INDICADOR DE NEGATIVO                EMU07240
GR6  D    0885 2303 0010 0011 0000 0011 0725         MSGR              GR6,D             D<--INDICADOR DE ZERO                    EMU07250
     D    0886 11A3 0001 0001 1010 0011 0726         MAR    OR         D                                                          EMU07260
0000 B    0887 C035 1100 0000 0011 0101 0727         MID               X'000',B                                                   EMU07270
     0A56 0888 A296 1010 0010 1001 0110 0728         JDS               LEREG                                                      EMU07280
     LM2  0889 0015 0000 0000 0001 0101 0729         MOLM              2                                                          EMU07290
0400 B    088A D001 1101 0000 0000 0001 0730         MID               X'400',B          B<--'0000010000000000'                   EMU07300
LM2       088B 0014 0000 0000 0001 0100 0731         MILM              2                                                          EMU07310
     D    088C 11A3 0001 0001 1010 0011 0732         MAR    OR         D                                                          EMU07320
0000 B    088D C035 1100 0000 0011 0101 0733         MID               X'000',B          D<--ST OR B, P(5)<--1 (GT)               EMU07330
     LM1  088E 0000 0000 0000 0000 1101 0734         MOLM              1                                                          EMU07340
     0A50 088F A294 1010 0010 1001 0100 0735         JDS               GRAVRG                                                     EMU07350
                                        0736  * SE RECEBE 1*****************************************************                  EMU07360
0000 B    0890 C035 1100 0000 0011 0101 0737         MID               X'000',B                                                   EMU07370
     0A56 0891 A296 1010 0010 1001 0110 0738         JDS               LEREG                                                      EMU07380
     LM2  0892 0015 0000 0000 0001 0101 0739         MOLM              2                                                          EMU07390
0200 B    0893 C801 1100 1000 0000 0001 0740         MID               X'200',B                                                   EMU07400
LM2       0894 0014 0000 0000 0001 0100 0741         MILM              2                                                          EMU07410
     D    0895 11A3 0001 0001 1010 0011 0742         MAR    OR         D                 P(4)<---1                       (GE)     EMU07420
     08AC 0896 004B 0000 0000 0100 1011 0743         JRU               GTZERO+9          IF P/'GTZERO+9'                          EMU07430
0000 B    0897 C035 1100 0000 0011 0101 0744  GTZERO MID               X'000',B                                                   EMU07440
     0A56 0898 A296 1010 0010 1001 0110 0745         JDS               LEREG                                                      EMU07450
     LM2  0899 0015 0000 0000 0001 0101 0746         MOLM              2                                                          EMU07460
00FB B    089A C3ED 1100 0011 1110 1101 0747         MID               X'0FB',B                                                   EMU07470
B    B    089B 310A 0011 0001 0000 1010 0748         MSRR   X          B,B               B<---'1111101100000000'                  EMU07480
00FF D    089C C3FE 1100 0011 1111 1110 0749         MID               X'0FF',D          D<---'00FF'                              EMU07490
     B    089D 1222 0001 0010 0010 0010 0750         MAR    +0         B                 B<--'1111101111111111'                   EMU07500
LM2       089E 0014 0000 0000 0001 0100 0751         MILM              2                                                          EMU07510
```

```
                                                           C  .  1    .    2    .    3    .    4    .    5    .    6    .    7    .    8
ORIG  DEST  LOCN  INST  CONFIGURACAO  BITS    NINS   ROTULO OPER S M1/M2  C1 C2 ORIGEM/DESTINO   COMENTARIO                         ID/SEQ
      D     0B9F  10A3  0001 0000 1010 0011   0752          MAR    AND           D                 D<--ST AND B, D(S)<--D (GT)EMU07520
000D  B     0BA0  C035  1100 0000 0011 0101   0753          MID                  X'000',B                                       EMU07530
      LM1   0BA1  000D  0000 0000 0000 1101   0754          MOLM                 1                                            • EMU07540
      0A54  0BA2  A294  1010 0010 1001 0100   0755          JDS                  GRAVRG                                         EMU07550
STK   RPC   0BA3  3307  0011 0011 0000 0111   0756          MSRR                 STK,RPC           RETORNA 'FIM DA ARITH'       EMU07560
                                              0757   ------------------INSTRUCAO OPK--------------------------------------------EMU07570
            0BA4                              0758          BND                                                                 EMU07580
GR1   B     0BA4  2032  0010 0000 1000 0010   0759   OPK    MSGR                 GR1,B             B<--'INSTRUCAO'              EMU07590
      0CEC  0BA5  A333  1010 0011 0011 1011   0760          JDS                  ROTMON                                         EMU07600
0E23        0BA6  0E23  0000 1110 0010 0011   0761          DC                   CTOPK                                          EMU07610
B     0BAA  0BA7  9102  1001 0001 0000 0010   0762          TST          00      B,OPNIM           SE B(0)= 0 IR P/'OPNIM'      EMU07620
GR2   D     0BA8  2103  0010 0001 0000 0011   0763          MSGR                 GR2,D             D<--IMEDIATO                 EMU07630
      0BAC  0BA9  0013  0000 0000 0001 0011   0764          JRU                  OPNIM+2                                        EMU07640
GR1   B     0BAA  40FA  0100 0000 1111 1010   0765   OPNIM  MOGR   4  XSR        GR1,B             B<--RD                       EMU07650
      0A97  0BAB  A296  1010 0010 1001 0110   0766          JDS                  LEREG                                          EMU07660
D     GR4   0BAC  21C4  0010 0001 1100 0100   0767          MSRG                 D,GR4             GR4<--RD (OU IM)             EMU07670
GR1   B     0BAD  4CE2  0100 0000 1110 0010   0768          MOGR   4             GR1,B             B<--RS                       EMU07680
0BB0  D     0BAE  F0C2  1111 1111 1100 0010   0769          MID                  TABOPK,D          D<---'TABOPK'                EMU07690
      RPC   0BAF  1227  0001 0010 0010 0111   0770          MAR    +D            RPC               IR P/'TABOPK+RS'             EMU07700
      0BB4  0BB0  B2ED  1011 0010 1110 1101   0771   TABOPK JRU                  OPKZER                                         EMU07710
      0BC0  0BB1  B2F0  1011 0010 1111 0000   0772          JRU                  OPKUM                                          EMU07720
      0BC4  0BB2  B2F1  1011 0010 1111 0001   0773          JRU                  OPKNEG                                         EMU07730
            0BB4                              0774          BND                                    *****CONSTANTE ZERO*******   EMU07740
      0CEC  0BB4  A333  1010 0011 0011 1011   0775   OPKZER JDS                  ROTMON                                         EMU07750
0E26        0BB5  0E26  0000 1110 0010 0110   0776          DC                   CTKZER                                         EMU07760
000D  D     0BB6  C002  1100 0000 0000 0010   0777          MID                  X'000',D          D<---'0000'                  EMU07770
D     GR3   0BB7  21C3  0010 0001 1100 0011   0778          MSRG                 D,GR3             GR4<--CONSTANTE <*OPKZER+1*EMU07780
      0A99  0BB8  A2A6  1010 0010 1010 0110   0779          JDS                  OHER                                           EMU07790
GR1   B     0BB9  40FA  0100 0000 1111 1010   0780          MOGR   4  XSR        GR1,B             B<--RD                       EMU07800
GR3   D     0BBA  2133  0010 0001 1000 0011   0781          MSGR                 GR3,D             D<---RESULTADO               EMU07810
      LM1   0BBB  000D  0000 0000 0000 1101   0782          MOLM                 1               LM1<---RESULTADO               EMU07820
      0A5C  0BBC  A294  1010 0010 1001 0100   0783          JDS                  GRAVRG            D<---RESULTADO               EMU07830
      0B0D  0BBD  A2C2  1010 0010 1100 0010   0784          JDS                  ARITH                                          EMU07840
      073C  0BBE  0247  1011 0010 0110 1111   0785          JRU                  TESTE                                          EMU07850
            0BC0                              0786          BND                                    **********CONSTANTE UM*****EMU07860
      0CEC  0BC0  A333  1010 0011 0011 1011   0787   OPKUM  JDS                  ROTMON                                         EMU07870
0E24        0BC1  0E24  0000 1110 0010 0100   0788          DC                   CTKUM                                          EMU07880
0001  D     0BC2  C006  1100 0000 0000 0110   0789          MID                  X'001',D          D<---'0001'                  EMU07890
      0BB7  0BC3  0793  0000 0111 1001 1011   0790          JRU                  OPKZER+3          IR P/'OPKZER+3'              EMU07900
            0BC4                              0791          BND                                    *******CONSTANTE -1********  EMU07910
00FF  B     0BC4  C0FD  1100 0011 1111 1101   0792   OPKNEG MID                  X'0FF',B          D<--'0000000011111111'       EMU07920
      0CEC  0BC5  A333  1010 0011 0011 1011   0793          JDS                  ROTMON                                         EMU07930
0E25        0BC6  0E25  0000 1110 0010 0101   0794          DC                   CTKUMN                                         EMU07940
B     B     0BC7  310A  0011 0001 0000 1010   0795          MSRR   X             B,B               B<--'1111111100000000'       EMU07950
00FF  D     0BC8  C3FE  1100 0011 1111 1110   0796          MID                  X'0FF',D          D<--'0000000011111111'       EMU07960
      D     0BC9  1223  0001 0010 0010 0011   0797          MAR    +D            D                 D<--'FFFF'                   EMU07970
      0BB7  0BCA  0763  0000 0111 0110 0011   0798          JRU                  OPKZER+3          IR P/ 'OPKZER+3'             EMU07980
                                              0799   *-------------INSTRUCAO IE--INSERE/EXTRAI BYTE----------------------------EMU07990
            0BCC                              0800          BND                                                                 EMU08000
GR1   B     0BCC  2032  0010 0000 1000 0010   0801   IE     MSGR                 GR1,B             B<--INSTRUCAO                EMU08010
      0CEC  0BCD  A333  1010 0011 0011 1011   0802          JDS                  ROTMON                                         EMU08020
0E27        0BCE  0E27  0000 1110 0010 0111   0803          DC                   CTIE                                           EMU08030
B     0BD2  0BCF  9102  1001 0001 0000 0010   0804          TST          00      B,IENIM           SE B(0)=0 IR P/'IENIM'       EMU08040
GR2   D     0BD0  2103  0010 0001 0000 0011   0805          MSGR                 GR2,D             D<--IMEDIATO                 EMU08050
      0BD4  0BD1  0013  0000 0000 0001 0011   0806          JRU                  IENIM+2           IR P/'IENIM+2'               EMU08060
GR1   B     0BD2  40FA  0100 0000 1111 1010   0807   IENIM  MOGR   4  XSR        GR1,B             B<--RD                       EMU08070
      0A58  0BD3  A296  1010 0010 1001 0110   0808          JDS                  LEREG                                          EMU08080
D     GR4   0BD4  21C4  0010 0001 1100 0100   0809          MSRG                 D,GR4             GR4<--(RD)OU(IM)             EMU08090
```

```
                                          C  .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG  DEST  LOCN  INST  CONFIGURACAO  BITS  NINS  ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO              ID/SEQ
GR1   B     0BD5  40E2  0100 0000 1110 0010  0810         MOGR   4             GR1,B          B<--RS                  EMU08100
      0A58  0BD6  A296  1010 0010 1001 0110  0811         JDS               .  LEREG                                   EMU08110
D     GR3   0BD7  21C3  0010 0001 1100 0011  0812         MSRC                 D,GR3          GR3<--RS                EMU08120
GR1   B     0BD8  40EA  0100 0000 1110 1010  0813         MOGR   4  X          GR1,B                                  EMU08130
0007  D     0BD9  CD1E  1100 0000 0001 1110  0814         MID                  X'007',D                               EMU08140
      B     0BDA  10A2  0001 0000 1010 0010  0815         MAR    AND           B              B<--CODIGO DA OPERACAO(UEI)EMU08150
0BDE  D     0BDB  EF7A  1110 1111 0111 1010  0816         MID                  TABIE,D        D<--TABIE(ENDERECO)     EMU08160
      B     0BDC  1222  0001 0010 0010 0010  0817         MAR    +0            B              B<--TABIE+COD OPER(IE)  EMU08170
B     MPC   0BDD  3107  0011 0001 0000 0111  0818         MSRR                 B,MPC          MPC<--ENDERECO DA ROTINA EMU08180
      0BE0  0BDE  B2F8  1011 0010 1111 1000  0819  TABIE  JDU                  INSERE                                 EMU08190
      0BF0  0BDF  B2FC  1011 0010 1111 1100  0820         JDU                  EXTRAI                                 EMU08200
                                             0821  *----------------------ROTINAS DAS OPERACOES(UEI)------------------------EMU08210
                                             0822  *-------------------------OPERACAO INSERE BYTE--------------------------EMU08220
            0BE0                             0823         BND                                                         EMU08230
0000  B     0BE0  C035  1100 0000 0011 0101  0824  INSERE MID                  X'000',B                               EMU08240
      0CEC  0BE1  A333  1010 0011 0011 0011  0825         JDS                  ROTMON                                 EMU08250
0E28        0BE2  0E28  0000 1110 0010 1000  0826         DC                   CTINSE                                 EMU08260
      0A58  0BE3  A296  1010 0010 1001 0110  0827         JDS                  LEREG                                  EMU08270
D     0BE9  0BE4  91F4  1001 0001 1111 0100  0828         TST          70      D,INDO         SE IND=0 IR P/'INDO'    EMU08280
GR4   B     0BE5  4242  0100 0010 0100 0010  0829         MOGR   8             GR4,B          B<--'00GR4(8.15)'       EMU08290
                                             0830  *                           GR4 IS REGISTRADOR DESTINO OU IMEDIATO EMU08300
GR3   D     0BE6  41C3  0100 0001 1100 0011  0831         MOGR   8             GR3,D          D<--'00GR3(8.15)00'     EMU08310
                                             0832  *                           GR3 IS REGISTRADOR FONTE               EMU08320
D     D     0BE7  3188  0011 0001 1000 1011  0833         MSRR   X             D,D            D<--'GR3(8.15)00'       EMU08330
      0BEE  0BE8  0023  0000 0000 0010 1011  0834         JRU                  INDO+5         IR P/'INDO+5'           EMU08340
00FF  B     0BE9  C3FD  1100 0011 1111 1101  0835  INDO   MID                  X'0FF',B       B<--'00FF'              EMU08350
B     B     0BEA  310A  0011 0001 0000 1010  0836         MSRR   X             B,B            B<--'FF00'              EMU08360
GR4   D     0BEB  2203  0010 0010 0000 0011  0837         MSGR                 GR4,D          D<--GR4                 EMU08370
      B     0BEC  10A2  0001 0000 1010 0010  0838         MAR    AND           B              B<--GR4(0.7)00'         EMU08380
GR3   D     0BED  41C3  0100 0001 1100 0011  0839         MOGR   3             GR3,D          D<--GR3                 EMU08390
      D     0BEE  1223  0001 0010 0010 0011  0840         MAR    +0            D              D<--RD I RS      *IND+5*EMU08400
      0BFC  0BEF  0063  0000 0000 0110 0011  0841         JRU                  EIND0+1        IR P/'EIND0+1'          EMU08410
                                             0842  *------------------------OPERACAO EXTRAI BYTE-------------------------EMU08420
            0BF0                             0843         BND                                                         EMU08430
0000  B     0BF0  C035  1100 0000 0011 0101  0844  EXTRAI MID                  X'000',B       D<--LM(13), D<--ST      EMU08440
      0CEC  0BF1  A333  1010 0011 0011 0011  0845         JDS                  ROTMON                                 EMU08450
0E29        0BF2  0E29  0000 1110 0010 1001  0846         DC                   CTEXTR                                 EMU08460
      0A58  0BF3  A296  1010 0010 1001 0110  0847         JDS                  LEREG                                  EMU08470
D     0BFB  0BF4  91F6  1001 0001 1111 0110  0848         TST          70      D,EIND0        SE IND=0 IR P/'EIND0'   EMU08480
00FF  B     0BF5  C3FD  1100 0011 1111 1101  0849         MID                  X'0FF',B       B<--'00FF'              EMU08490
B     B     0BF6  310A  0011 0001 0000 1010  0850         MSRR   X             B,B            B<--'FF00'              EMU08500
GR3   D     0BF7  2133  0010 0001 1000 0011  0851         MSGR                 GR3,D          D<--RS                  EMU08510
      D     0BF8  10A3  0001 0000 1010 0011  0852         MAR    AND           D              D<--'GR3(0.7)00'        EMU08520
D     D     0BF9  3188  0011 0001 1000 1011  0853         MSRR   X             D,D            D<--RD E RS  (IND=0)    EMU08530
      0BFC  0BFA  0063  0000 0000 0110 1011  0854         JRU                  EIND0+1        IR P/'EIND0+1'          EMU08540
GR3   D     0BFB  41C3  0100 0001 1100 0011  0855  EIND0  MOGR   3             GR3,D          D<--RD E RS (IND=0)     EMU08550
GR1   B     0BFC  40FA  0100 0000 1111 1010  0856         MOGR   4  XSR        GR1,B          B<--RD         *FIND1+1*EMU08560
      LM1   0BFD  0000  0000 0000 0000 1101  0857         MOLM                 1                                      EMU08570
      0A5C  0BFE  A294  1010 0010 1001 0100  0858         JDS                  GRAVRG                                 EMU08580
      093C  0BFF  B24F  1011 0010 0100 1111  0859         JDU                  TESTE                                  EMU08590
                                             0860  *------------------------SETAR SE IGUAL A ZERO------------------------EMU08600
                                             0861  *                          ***(SEQ)***                             EMU08610
            0C00                             0862         BND                                                         EMU08620
      0CEC  0C00  A333  1010 0011 0011 1011  0863  SEQ    JDS                  ROTMON                                 EMU08630
0E2A        0C01  0E2A  0000 1110 0010 1010  0864         DC                   CTSEQ                                  EMU08640
      0C28  0C02  A30A  1010 0011 0000 1010  0865         JDS                  GERAL          CALL 'GERAL'            EMU08650
      B     0C03  1322  0001 0011 0010 0010  0866         MAR    -1            B              B<---RD-RS(OU IM)       EMU08660
S     0C10  0C04  9475  1001 0100 0111 1011  0867         TST          71      S,ZERARD                               EMU08670
```

```
                                                     C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCN INST CONFIGURACAO        BITS  NINS   ROTULO OPER S M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                    ID/SEQ
     0C08 0C05 B302 1011 0011 0000 0010       0868          JOU                SETRO                                         EMU08680
          0C08                                0869          BND                                                              EMU08690
00FF B    0C08 C3FD 1100 0011 1111 1101       0870   SETRO  MID                X'0FF',B                                      EMU08700
B    B    0C09 313A 0011 0001 0000 1010       0871          MSRR   X           B,B            I----------------------------  EMU08710
00FF D    0C0A C3FE 1100 0011 1111 1110       0872          MID                X'0FF',D       I          D<--'FFFF'          EMU08720
     D    0C0B 1223 0001 0010 0010 0011       0873          MAR    +0          D              I                              EMU08730
     0C11 0C0C 0023 0000 0000 0010 0011       0874          JRU                ZERARD+1       I----------------------------  EMU08740
          0C10                                0875          BND                                                              EMU08750
0000 D    0C10 C002 1100 0000 0000 0010       0876   ZERARD MID                X'000',D                                      EMU08760
     LM1  0C11 000D 0000 0000 0000 1101       0877          MOLM               1              LM(1)<--0 OU -1                EMU08770
GR1  B    0C12 40FA 0100 0000 1111 1010       0878          MOGR   4  XSR      GR1,B          B<---RD                        EMU08780
     0A50 0C13 A294 1010 0010 1001 0100       0879          JDS                GRAVRG                                        EMU08790
     093C 0C14 B24F 1011 0010 0100 1111       0880          JOU                TESTE                                         EMU08800
                                              0881   *-------------------SETAR SE MAIOR QUE ZERO---------------------------- EMU08810
                                              0882   *                         ***(SGT)***                                   EMU08820
          0C18                                0883          BND                                                              EMU08830
     0CEC 0C18 A333 1010 0011 0011 1011       0884   SGT    JDS                ROTMON                                        EMU08840
0E28      0C19 0E28 0000 1110 0010 1011       0885          DC                 CTSGT                                         EMU08850
     0C28 0C1A A30A 1010 0011 0000 1010       0886          JDS                GERAL          CALL 'GERAL'                   EMU08860
     B    0C1B 1322 0001 0011 0010 0010       0887          MAR    -1          B                                             EMU08870
S    0C1E 0C1C 9471 1001 0100 0111 0001       0888          TST          71    S,TSTNEG                                      EMU08880
     0C10 0C1D B304 1011 0011 0000 0100       0889          JOU                ZERARD                                        EMU08890
B    0C10 0C1E 9001 1001 1101 0000 0001       0890   TSTNEG TST          01    B,ZERARD                                      EMU08900
     0C08 0C1F B302 1011 0011 0000 0010       0891          JOU                SETRO                                         EMU08910
                                              0892   *-------------------SETAR SE MAIOR OU IGUAL A ZERO--------------------- EMU08920
                                              0893   *                         ***(SGE)***                                   EMU08930
     0CEC 0C20 A333 1010 0011 0011 1011       0894   SGE    JDS                ROTMON                                        EMU08940
0E2C      0C21 0E2C 0000 1110 0010 1100       0895          DC                 CTSGE                                         EMU08950
     0C28 0C22 A30A 1010 0011 0000 1010       0896          JDS                GERAL          CALL 'GERAL'                   EMU08960
     B    0C23 1322 0001 0011 0010 0010       0897          MAR    -1          B                                             EMU08970
B    0C26 0C24 9501 1001 0101 0000 0001       0898          TST          01    B,XZERA                                       EMU08980
     0C08 0C25 B302 1011 0011 0000 0010       0899          JOU                SETRO                                         EMU08990
     0C10 0C26 B304 1011 0011 0000 0100       0900   XZERA  JOU                ZERARD                                        EMU09000
                                              0901   *-------------------SUBROTINA GERAL------------------------------------ EMU09010
          0C28                                0902          BND                                                              EMU09020
GR1  B    0C28 40FA 0100 0000 1111 1010       0903   GERAL  MOGR   4  XSR      GR1,B           B<--RD                        EMU09030
     0A55 0C29 A296 1010 0010 1001 0110       0904          JDS                LERED          D<--(RD)                       EMU09040
D    GR4  0C2A 2104 0010 0001 0000 0100       0905          MSRG               D,GR4          GR4<--(RD)                     EMU09050
GR1  B    0C2B 2082 0010 0000 1000 0010       0906          MSGR               GR1,B          B<---INSTRUCAO                 EMU09060
B    0C31 0C2C 9504 1001 0101 0000 0100       0907          TST          01    B,GERIM        IR P/'GERIM'(IMEDIATO)         EMU09070
GR1  B    0C2D 40E2 0100 0000 1110 0010       0908          MOGR   4           GR1,B           B<--RS                        EMU09080
     0A56 0C2E A296 1010 0010 1001 0110       0909          JDS                LEREG          D<--(RS)                       EMU09090
D    B    0C2F 3182 0011 0001 1000 0010       0910          MSRR               D,B            B<--(RS)                       EMU09100
     0C32 0C30 000B 0000 0000 0000 1011       0911          JRU                GERIM+1                                       EMU09110
GR2  B    0C31 2102 0010 0001 0000 0010       0912   GERIM  MSGR               GR2,B          B<--IMEDIATO                   EMU09120
GR4  D    0C32 2203 0010 0010 0000 0011       0913          MSGR               GR4,D          D<--(RD)                       EMU09130
STK  MPC  0C33 3307 0011 0011 0000 0111       0914          MSRR               STK,MPC                                       EMU09140
                                              0915   *-------------------BRANCH SE FALSO(BF)-------------------------------- EMU09150
          0C34                                0916          BND                                                              EMU09160
GR1  B    0C34 40E2 0100 0000 1110 0010       0917   BF     MOGR   4           GR1,B          B<--RS                         EMU09170
     0CEC 0C35 A333 1010 0011 0011 1011       0918          JDS                ROTMON                                        EMU09180
0E2D      0C36 0E2D 0000 1110 0010 1101       0919          DC                 CTBF                                          EMU09190
     0A58 0C37 A296 1010 0010 1001 0110       0920          JDS                LEREG                                         EMU09200
0000 B    0C38 C001 1100 0000 0000 0001       0921          MID                X'000',B       B<---'0000'                    EMU09210
     D    0C39 1223 0001 0010 0010 0011       0922          MAR    +0          D              D<---RS+0                      EMU09220
S    0C3F 0C3A 9474 1001 0100 0111 0100       0923          TST          71    S,K1           SE S(7)=1 IR P/'K1'            EMU09230
GR1  B    0C3B 40FA 0100 0000 1111 1010       0924          MOGR   4  XSR      GR1,B           B<--RD                        EMU09240
GR2  D    0C3C 2103 0010 0001 0000 0011       0925          MSGR               GR2,D          D<---IMEDIATO                  EMU09250
```

```
ORIG DEST LOCN INST CONFIGURACAO        BITS NINS  C . 1 . 2 . 3 . 4 . 5 . 6 . 7 . 8
                                                   ROTULO OPER S M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO               ID/SEQ
     LM1  0C3D 000D 0000 0000 0000 1101 0926         MOLM        1                                                   EMU09260
     0A5D 0C3E A294 1010 0010 1001 0100 0927         JDS         GRAVRG                                              EMU09270
     093C 0C3F B24F 1011 0010 0100 1111 0928  K1     JDU         TESTE            IR P/'TESTE'                       EMU09280
                                         0929  *-------------------ROTACAO COM CARRY(RC)-------------------------------EMU09290
          0C40                           0930         END                                                             EMU09300
GR1  B    0C40 2032 0010 0000 1000 0010 0931  RC     MSGR        GR1,B            S<---INSTRUCAO                     EMU09310
     0CEC 0C41 A33B 1010 0011 0011 1011 0932         JDS         ROTMON                                              EMU09320
0E2E      0C42 0E2E 0000 1110 0010 1110 0933         DC          CTRC                                                EMU09330
B    0C45 0C43 9101 1001 0001 0000 0001 0934         TST      00 B,NAOIM                                             EMU09340
     0C48 0C44 001B 0000 0000 0001 1011 0935         JRU         NAOIM+3                                             EMU09350
GR1  B    0C45 40E2 0100 0000 1110 0010 0936  NAOIM  MOGR  4     GR1,B            IR P/'NAOIM+3'                     EMU09360
     0A5B 0C46 A296 1010 0010 1001 0110 0937         JDS         LEREG             B<--RS                            EMU09370
D    GR2  0C47 21C2 0010 0001 1100 0010 0938         MSRB        D,GR2            D<--(RS)                           EMU09380
                                         0939  * GR2 CONTEM IMEDIATO OU RS         GR2<--(RS)                         EMU09390
GR1  B    0C48 40EA 0100 0000 1110 1010 0940         MOGR  4 X   GR1,B            B<--BIT(12) E OP   *NAOIM+3*EMU09400
0007 D    0C49 C01E 1100 0000 0001 1110 0941         MID         X'007',D         D<--'0007'                         EMU09410
     B    0C4A 10A2 0001 0000 1010 0010 0942         MAR   AND   B                B<--OP                             EMU09420
0C4E D    0C4B F13A 1111 0001 0011 1010 0943         MID         TABROT,D                                            EMU09430
     B    0C4C 1222 0001 0010 0010 0010 0944         MAR   +0    B                                                   EMU09440
B    MPC  0C4D 3107 0011 0001 0000 0111 0945         MSRR        B,MPC                                               EMU09450
     0C54 0C4E B315 1011 0011 0001 0101 0946  TABROT JDU         ESQ                                                 EMU09460
          0C4F 0000 0000 0000 0000 0000 0947         NOP                                                             EMU09470
     0C7C 0C50 B31F 1011 0011 0001 1111 0948         JDU         DIR                                                 EMU09480
          0C51 0000 0000 0000 0000 0000 0949         NOP                                                             EMU09490
     0C88 0C52 B322 1011 0011 0010 0010 0950         JDU         MOVE                                                EMU09500
          0C54                           0951         END                                                             EMU09510
GR2  B    0C54 2122 0010 0001 0010 0010 0952  ESQ    MSGR  SL    GR2,B                                               EMU09520
     0CEC 0C55 A33B 1010 0011 0011 1011 0953         JDS         ROTMON                                              EMU09530
0E2F      0C56 0E2F 0000 1110 0010 1111 0954         DC          CTROLC                                              EMU09540
S    GR3  0C57 2043 0010 0000 0100 0011 0955         MSRB        S,GR3            GUARDA S EM GR3                    EMU09550
                                         0956  *--------B<==GR2 ROTADO C/ZERO----S(4)<==GR2(0)----R(15)<==0----------- EMU09560
[illegible] GR4 0C58 2144 0010 0001 0100 0100 0957   MSRB        B,GR4            GR4<==RS ROTADO C/ZERO             EMU09570
0001 D    0C59 C006 1100 0000 0000 0110 0958         MID         X'001',D         ENTRA UM                           EMU09580
     LM5  0C5A 002D 0000 0000 0010 1101 0959         MOLM        5                LM5<=='0001'                       EMU09590
                                         0960  *--------BUSCA REGISTRADOR DE ESTADOS(ST) E TESTA CARRY----------------- EMU09600
000D B    0C5B C035 1100 0000 0011 0101 0961         MID         X'00D',B                                            EMU09610
     0A5B 0C5C A296 1010 0010 1001 0110 0962         JDS         LEREG                                               EMU09620
D    0C61 0C5D 91A3 1001 0001 1010 0011 0963         TST   [illegible] D,SAI   D<==(ST)                             EMU09630
LM5       0C5E 002C 0000 0000 0010 1100 0964         MILM        5                SE CARRY=0 ENTAO SAI               EMU09640
GR4  B    0C5F 2202 0010 0010 0000 0010 0965         MSGR        GR4,B            D<==LM5                            EMU09650
     GR4  0C60 1184 0001 0001 1000 0100 0966         MAG   OR    GR4              R<==RS OU IM ROTADO C/ZERO         EMU09660
GR3  B    0C61 2132 0010 0001 1000 0010 0967  SAI    MSGR        GR3,B            GR4<--RS OU IM ROTADO C/UM*EMU09670
B    0C6D 0C62 954A 1001 0101 0100 1010 0968         TST      41 B,ESQUM          B <--S                            *EMU09680
000D B    0C63 C035 1100 0000 0011 0101 0969         MID         X'00D',B         SE S(4)=1 IR P  'ESQUM'            EMU09690
     0A5B 0C64 A296 1010 0010 1001 0110 0970         JDS         LEREG                                               EMU09700
     LM2  0C65 0015 0000 0000 0001 0101 0971         MOLM        2                                                   EMU09710
00DF B    0C66 C37D 1100 0011 0111 1101 0972         MID         X'0DF',B         LM(2)<--ST                         EMU09720
00FF D    0C67 C3FE 1100 0011 1111 1110 0973         MID         X'0FF',D                                            EMU09730
B    B    0C68 310A 0011 0001 0000 1010 0974         MSRR  X     B,B                                                 EMU09740
     B    0C69 1222 0001 0010 0010 0010 0975         MAR   +0    B                                                   EMU09750
LM2       0C6A 0014 0000 0000 0001 0100 0976         MILM        2                R<--'0FFF'                         EMU09760
     D    0C6B 10A3 0001 0000 1010 0011 0977         MAR   AND   D                D<---ST                            EMU09770
     0C74 0C6C 0C3B 0000 0000 0011 1011 0978         JRU         ESQUM+7          D<--R AND ST(CARRY RESETADOEMU09780
000D B    0C6D C035 1100 0000 0011 0101 0979  ESQUM  MID         X'000',B                                            EMU09790
     0A5B 0C6E A296 1010 0010 1001 0110 0980         JDS         LEREG                                               EMU09800
     LM2  0C6F 0015 0000 0000 0001 0101 0981         MOLM        2                                                   EMU09810
0020 B    0C70 C081 1100 0000 1000 0001 0982         MID         X'020',B                                            EMU09820
B    B    0C71 310A 0011 0001 0000 1010 0983         MSRR  X     B,B              S<--'2000'                         EMU09830
```

```
                                                         C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG DEST LOCM INST CONFIGURACAO        NINS  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                   ID/SEQ
LM2       0C72 0014 0000 0000 0001 0100 0984         MILM           2                                            EMU09840
     D    0C73 11A3 0001 0001 1010 0011 0985         MAR    OR      D               D<--B OR ST(CARRY SETADO)    EMU09850
000D B    0C74 C035 1100 0000 0011 0101 0986         MID            X'00D',B                                     EMU09860
     LM1  0C75 000D 0000 0000 0000 1101 0987         MOLM           1                                            EMU09870
     0A50 0C76 A294 1010 0010 1001 0100 0988         JDS            GRAVRG                                       EMU09880
GR1  B    0C77 40FA 0100 0000 1111 1010 0989         MOGR   4  XSR  GR1,B           B<---RD          *ESDUM+5*   EMU09890
GR4  D    0C78 2203 0010 0010 0000 0011 0990         MSGR           GR4,D           D<---ROT(RS) OU ROT(IM)      EMU09900
     LM1  0C79 000D 0000 0000 0000 1101 0991         MOLM           1                                            EMU09910
     0A50 0C7A A294 1010 0010 1001 0100 0992         JDS            GRAVRG                                       EMU09920
     093C 0C7B B24F 1011 0010 0100 1111 0993         JDU            TESTE                                        EMU09930
          0C7C                          0994         BND                                                         EMU09940
GR2  B    0C7C 212A 0010 0001 0010 1010 0995  DIR    MSGR   SR      GR2,B           S(4)<--GR2(15), GR2(0)<--0   EMU09950
     0CEC 0C7D A33B 1010 0011 0011 1011 0996         JDS            ROTMON                                       EMU09960
0E30      0C7E 0E30 0000 1110 0011 0000 0997         DC             CTRORC                                       EMU09970
S    GR3  0C7F 2343 0010 0011 0100 0011 0998         MSRG           S,GR3           GUARDA S EM GR3              EMU09980
B    GR4  0C80 2144 0010 0001 0100 0100 0999         MSRG           B,GR4           GR4 <--B ROTADO C/ZERO       EMU09990
0080 D    0C81 C203 1100 0010 0000 0011 1000         MID            X'080',D                                     EMU10000
D    D    0C82 318B 0011 0001 1000 1011 1001         MSRR   X       D,D             D<--'F000' ENTRA UM          EMU10010
     LM5  0C83 002D 0000 0000 0010 1101 1002         MOLM           5               LM(5)<--'8000'               EMU10020
     0C58 0C84 06B3 0000 0110 1011 0011 1003         JRU            ESQ+7                                        EMU10030
          0C88                          1004         BND                                                         EMU10040
GR2  B    0C88 2102 0010 0001 0000 0010 1005  MOVE   MSGR           GR2,B           B<--IM OU RS                 EMU10050
     0CEC 0C89 A33B 1010 0011 0011 1011 1006         JDS            ROTMON                                       EMU10060
0E31      0C8A 0E31 0000 1110 0011 0001 1007         DC             CTMOVE                                       EMU10070
B    GR4  0C8B 2144 0010 0001 0100 0100 1008         MSRG           B,GR4           GR4<---B                     EMU10080
     0C77 0C8C 0753 0000 0111 0101 0011 1009         JRU            ESDUM+10                                     EMU10090
                                        1010  *-----------------------INSTRUCAO STORE---------------------------------EMU10100
          0C90                          1011         BND                                                         EMU10110
GR1  B    0C90 2082 0010 0000 1000 0010 1012  STORE  MSGR           GR1,B           B<--INSTRUCAO                EMU10120
     0CEC 0C91 A33B 1010 0011 0011 1011 1013         JDS            ROTMON                                       EMU10130
0E32      0C92 0E32 0000 1110 0011 0010 1014         DC             CTSTOR                                       EMU10140
B    0C96 0C93 9102 1001 0001 0000 0010 1015         TST         OL B,STORD                                      EMU10150
GR2  A    0C94 2104 0010 0001 0000 0100 1016         MSGR           GR2,A           A<--IMEDIATO                 EMU10160
     0C99 0C95 001B 0000 0000 0001 1011 1017         JRU            STORD+3                                      EMU10170
B    B    0C96 617A 0110 0001 0111 1010 1018  STORD  MORR   4  XSR  B,B             B<--RD                       EMU10180
     0A58 0C97 A296 1010 0010 1001 0110 1019         JDS            LEREG                                        EMU10190
D    A    0C98 3184 0011 0001 1000 0100 1020         MSRR           D,A             A<--(RD)                     EMU10200
     0A40 0C99 A290 1010 0010 1001 0000 1021         JDS            AJENDR          A<==IM OU RD AJUST*STORD+3*  EMU10210
A    GR4  0C9A 2244 0010 0010 0100 0100 1022         MSRG           A,GR4           GR4<--ENDERECO AJUSTADO     *EMU10220
GR1  B    0C9B 40E2 0100 0000 1110 0010 1023         MOGR   4       GR1,B           B<--RS                       EMU10230
     0A58 0C9C A296 1010 0010 1001 0110 1024         JDS            LEREG           D<--(RS)                     EMU10240
GR4  A    0C9D 2204 0010 0010 0000 0100 1025         MSGR           GR4,A           A<--ENDERECO AJUSTADO        EMU10250
          0C9E 0816 0000 1000 0001 0110 1026         RSET           14                                           EMU10260
B    B    0C9F 3902 0011 1001 0000 0010 1027         MSRR   W       B,B             MEMP(A)<--(RS)               EMU10270
B    B    0CA0 3D02 0011 1101 0000 0010 1028         MSRR   S       B,B             ESPERA GRAVACAO              EMU10280
     093C 0CA1 B24F 1011 0010 0100 1111 1029         JDU            TESTE                                        EMU10290
                                        1030  *-----------------------INSTRUCAO PUSH----------------------------------EMU10300
          0CA4                          1031         BND                                                         EMU10310
GR1  B    0CA4 2082 0010 0000 1000 0010 1032  PUSH   MSGR           GR1,B           B<--INSTRUCAO                EMU10320
     0CEC 0CA5 A33B 1010 0011 0011 1011 1033         JDS            ROTMON                                       EMU10330
0E33      0CA6 0E33 0000 1110 0011 0011 1034         DC             CTPUSH                                       EMU10340
B    B    0CA7 617A 0110 0001 0111 1010 1035         MORR   4  XSR  B,B             B<--RD                       EMU10350
     0A58 0CA8 A296 1010 0010 1001 0110 1036         JDS            LEREG                                        EMU10360
D    A    0CA9 3184 0011 0001 1000 0100 1037         MSRR           D,A             A<--(RD)                     EMU10370
D    GR4  0CAA 21C4 0010 0001 1100 0100 1038         MSRG           D,GR4           GR4<--(RD) (OPER. DA OPER)   EMU10380
     0A40 0CAB A290 1010 0010 1001 0000 1039         JDS            AJENDR          A<--ENDERECO AJUSTADO        EMU10390
A    GR5  0CAC 2245 0010 0010 0100 0101 1040         MSRG           A,GR5           SALVA ENDERECO EM GR5        EMU10400
GR1  B    0CAD 40E2 0100 0000 1110 0010 1041         MOGR   4       GR1,B           B<--RS                       EMU10410
```

```
                                                         C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG  DEST  LOCN  INST  CONFIGURACAO         BITS  MINS  ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO             ID/SEQ
      0A58  0CAE  A296  1010 0010 1001 0110        1042         JDS                  LEREG                                  EMU10420
GR5   A     0CAF  2234  0010 0010 1000 0100        1043         MSGR                 GR5,A          DEVOLVE ENDERECO PARA A EMU10430
B     B     0CB0  3902  0011 1001 0000 0010        1044         MSRR 4               B,B            MEMP(A)<--RS            EMU10440
D     B     0CB1  3D02  0011 1101 0000 0010        1045         MSRR 5               B,D            MEMP(A)<--RS            EMU10450
0001  B     0CB2  C005  1100 0000 0000 0101        1046         MID                  X'0001',B      B<---'0001'             EMU10460
B     GR3   0CB3  2143  0010 0001 0100 0011        1047         MSRG                 B,GR3          GR3<--'0001'(OPER. DA OPER)EMU10470
      0A98  0CB4  A2A6  1010 0010 1010 0110        1048         JDS                  OPER                                   EMU10480
                                                   1049  * GR3 CONTEM RESULTADO**********                                       EMU10490
GR1   B     0CB5  40FA  0100 0000 1111 1010        1050         MOGR   4  XSP        GR1,B          B<---RD                 EMU10500
GR3   D     0CB6  2183  0010 0001 1000 0011        1051         MSGR                 GR3,D          D<--RESULT(RD OP 1)     EMU10510
      LM1   0CB7  0000  0000 0000 0000 1101        1052         MOLM                 1                                      EMU10520
      0A50  0CB8  A294  1010 0010 1001 0100        1053         JDS                  GRAVRG                                 EMU10530
GR1   B     0CB9  2082  0010 0000 1000 0010        1054         MSGR                 GR1,B          B<--INSTRUCAO           EMU10540
B     0CBF  0CBA  9104  1001 0001 0000 0100        1055         TST          03      B,SAIDA                                EMU10550
GR1   B     0CBB  40E2  0100 0000 1110 0010        1056         MOGR   4             GR1,B          B<--RS                  EMU10560
GR2   D     0CBC  2103  0010 0001 0000 0011        1057         MSGR                 GR2,D          D<--IMEDIATO            EMU10570
      LM1   0CBD  0000  0000 0000 0000 1101        1058         MOLM                 1                                      EMU10580
      0A50  0CBE  A294  1010 0010 1001 0100        1059         JDS                  GRAVRG                                 EMU10590
      093C  0CBF  324F  0011 0010 0100 1111        1060  SAIDA  JDU                  TESTE                                  EMU10600
                                                   1061  *---------------------------INSTRUCAO LOAD------------------------EMU10610
            0CC0                                   1062         END                                                         EMU10620
GR1   B     0CC0  2082  0010 0000 1000 0010        1063  LOAD   MSGR                 GR1,B          B<--INSTRUCAO           EMU10630
      0CEC  0CC1  A33B  1010 0011 0011 1011        1064         JDS                  ROTMON                                 EMU10640
0E34        0CC2  0E34  0000 1110 0011 0100        1065         DC                   CTLOAD                                 EMU10650
B     0CC7  0CC3  9503  1001 0101 0000 0011        1066         TST          01      B,LOADIM                               EMU10660
B     B     0CC4  6162  0110 0001 0110 0010        1067         MOPR   4             B,B            B<--RS                  EMU10670
      0A53  0CC5  A296  1010 0010 1001 0110        1068         JDS                  LEREG                                  EMU10680
D     GR2   0CC6  21C2  0010 0001 1100 0010        1069         MSRG                 D,GR2          GR2<---(RS)             EMU10690
GR2   A     0CC7  2104  0010 0001 0000 0100        1070  LOADIM MSGR                 GR2,A          A<--(RS) OU IMEDIATO    EMU10700
      0A40  0CC8  A290  1010 0010 1001 0000        1071         JDS                  AJENDR         A<--ENDERECO AJUSTADO   EMU10710
            0CC9  0A16  0000 1000 0001 0110        1072         RFET                 14                                     EMU10720
GR1   B     0CCA  2082  0010 0000 1000 0010        1073         MSGR                 GR1,B                                  EMU10730
B     B     0CCB  557A  0101 0101 0111 1010        1074         MOPR F 4  XSP        B,B            B<--RD, D<--MEMP(A)     EMU10740
B     B     0CCC  3002  0011 1101 0000 0010        1075         MSRR 5               B,B            ESPERA FIM LEITURA      EMU10750
      LM1   0CCD  0000  0000 0000 0000 1101        1076         MOLM                 1                                      EMU10760
      0A50  0CCE  A294  1010 0010 1001 0100        1077         JDS                  GRAVRG                                 EMU10770
      093C  0CCF  324F  0011 0010 0100 1111        1078         JDU                  TESTE                                  EMU10780
                                                   1079  *---------------------------INSTRUCAO POP-------------------------EMU10790
            0CD0                                   1080         END                                                         EMU10800
GR1   B     0CD0  2082  0010 0000 1000 0010        1081  POP    MSGR                 GR1,B          B<--INSTRUCAO           EMU10810
      0CEC  0CD1  A33B  1010 0011 0011 1011        1082         JDS                  ROTMON                                 EMU10820
0E35        0CD2  0E35  0000 1110 0011 0101        1083         DC                   CTPOP                                  EMU10830
B     0CD6  0CD3  9502  1001 0101 0000 0010        1084         TST          01      B,POPIX                                EMU10840
0001  D     0CD4  C006  1100 0000 0000 0110        1085         MID                  X'0001',D      D<--'0001'              EMU10850
D     GR2   0CD5  21C2  0010 0001 1100 0010        1086         MSRG                 D,GR2          GR2<---'0001'           EMU10860
GR2   D     0CD6  2103  0010 0001 0000 0011        1087  POPIX  MSGR                 GR2,D          D<--'0001' OU IM        EMU10870
D     GR3   0CD7  21C3  0010 0001 1100 0011        1088         MSRG                 D,GR3          GR3<-OPER. DA OPER(1 OU IM)EMU10880
B     B     0CD8  6162  0110 0001 0110 0010        1089         MOPR   4             B,B            B<--RS                  EMU10890
      0A53  0CD9  A296  1010 0010 1001 0110        1090         JDS                  LEREG                                  EMU10900
D     A     0CDA  3184  0011 0001 1000 0100        1091         MSRR                 D,A            A<---(RS)               EMU10910
D     GR4   0CDB  21C4  0010 0001 1100 0100        1092         MSRG                 D,GR4          GR4<--(RS)(OPER. DA OPER) EMU10920
      0A98  0CDC  A2A6  1010 0010 1010 0110        1093         JDS                  OPER                                   EMU10930
                                                   1094  *GR3 CONTEM RESULTADO*********************                        EMU10940
GR1   B     0CDD  40E2  0100 0000 1110 0010        1095         MOGR   4             GR1,B          B<--RS                  EMU10950
GR3   D     0CDE  2183  0010 0001 1000 0011        1096         MSGR                 GR3,D          D<--RESULT(RS OP GR3)   EMU10960
      LM1   0CDF  0000  0000 0000 0000 1101        1097         MOLM                 1                                      EMU10970
      0A50  0CE0  A294  1010 0010 1001 0100        1098         JDS                  GRAVRG                                 EMU10980
GR3   A     0CE1  2184  0010 0001 1000 0100        1099         MSGR                 GR3,A                                  EMU10990
```

```
ORIG DEST LOCN INST CONFIGURACAO        BITS NIVS  ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO   COMENTARIO              ID/SEQ
     0A40 0CE2 4297 1010 0010 1001 0000      1100         JDS                  AJENDF                                  EMU11000
          0CE3 0816 0000 1000 0001 0110      1101         RSET                 14                                      EMU11010
GR1  B    0CE4 44FA 0100 0100 1111 1010      1102         MOGR B 4   YSR       GR1,B         H<--DD D<--MEMR(A)        EMU11020
B    B    0CE5 3D02 0011 1101 0000 0010      1103         MSRR A               B,B           ESPERA FIM LEITURA        EMU11030
     LM1  0CE6 0000 0000 0000 0000 1101      1104         MOLM                 1                                       EMU11040
     CASO 0CE7 4294 1010 0010 1001 0100      1105         JDS                  GRAVRG                                  EMU11050
     093C 0CE8 524F 1011 0010 0100 1111      1106         JDN                  TESTE                                   EMU11060
                                             1107  *---------------------------ROTMON----------------------------------EMU11070
                                             1108  *ESTA ROTINA SOMA 1 NO CONTADOR CUJO EN DERECO E A PALAVRA SEGUINTE A   EMU11080
                                             1109  *INSTRUCAO DE CHAMADA                                                  EMU11090
                                             1110  *-----------------------------------------------------------------------EMU11100
          0CEC                               1111         BND                                                          EMU11110
STK  A    0CEC 3304 0011 0011 0000 0100      1112  ROTMON MSRR                 STK,A         A=END. PAL SEG INST CHAMADAEMU11120
A    D    0CED 1263 0001 0010 0110 0011      1113         MURR    +1           A,D                                     EMU11130
D    STK  0CEE 3186 0011 0001 1000 0110      1114         MSRR                 D,STK         STK RECEBE ENDERECO RETORNOEMU11140
          0CEF 0303 0000 1000 0000 0011      1115         CMRD                                                         EMU11150
D    A    0CF0 3184 0011 0001 1000 0100      1116         MSRR                 D,A           A=ENDERECO DO CONTADOR    EMU11160
          0CF1 0803 0000 1000 0000 0011      1117         CMRD                               D=CONTEUDO DO CONTADOR    -EMU11170
D    D    0CF2 11E3 0001 0001 1110 0011      1118         MURR    +1           D,D                                     --EMU11180
          0CF3 0802 0000 1000 0000 0010      1119         CMNT                               CONT=CONT+1               -EMU11190
STK  MPC  0CF4 3307 0011 0011 0000 0111      1120         MSRR                 STK,MPC                                 EMU11200
                                             1121  *---------------------------FETPAR----------------------------------EMU11210
                                             1122  *BUSCA DOS PARAMETROS NECESSARIOS A E/S EM DISQUETE. A PALAVRA 'FF5C'  EMU11220
                                             1123  *QUE CONTEM OS PARAMETROS FOI ENVIADA EM GR1 PELO TESTE                EMU11230
                                             1124  *-----------------------------------------------------------------------EMU11240
                                             1125  *BUSCA TRILHA NOS BITS 1 A 7 DE GR1                                    EMU11250
          0CF8                               1126         BND                                                          EMU11260
GR1  B    0CF8 203A 0010 0000 1000 1010      1127  FETPAR MSGR    X            GR1,B                                   EMU11270
007F D    0CF9 C1FE 1100 0001 1111 1110      1128         MID                  X'07F',D                                EMU11280
     D    0CFA 10A3 0001 0000 1010 0011      1129         MAR     AND          D                                       EMU11290
     2008 0CFB A502 1010 1000 0000 0010      1130         JDS                  SOD                                     EMU11300
0004      0CFC 0004 0000 1101 0000 0100      1131         DC                   TRIL1         TRIL1<---TRILHA           EMU11310
                                             1132  *BUSCA SETOR NOS BITS 8 A 12 DE GR1                                    EMU11320
GR1  D    0CFD 40C3 0100 0000 1100 0011      1133         MOGR    D            GR1,D                                   EMU11330
D    D    0CFE 31A3 0011 0001 1010 1011      1134         MSRR    SR           D,D                                     EMU11340
D    D    0CFF 31A3 0011 0001 1010 1011      1135         MSRR    SR           D,D                                     -EMU11350
D    D    0D00 31A3 0011 0001 1010 1011      1136         MSRR    SR           D,D                                     EMU11360
     2008 0D01 A802 1010 1000 0000 0010      1137         JDS                  SOD                                     EMU11370
0005      0D02 0005 0000 1101 0000 0101      1138         DC                   SET2          SET2<---SETOR             EMU11380
STK  MPC  0D03 3307 0011 0011 0000 0111      1139         MSRR                 STK,MPC                                 EMU11390
          0D04                               1140  TRIL1  DA                   1                                       EMU11400
          0D05                               1141  SET2   DA                   1                                       EMU11410
                                             1142  *---------------------------ZERACT----------------------------------EMU11420
                                             1143  *ESTA ROTINA ZERA TODOS OS 31 CONTADORES USADOS PARA CONTABILIZACAO DO  EMU11430
                                             1144  *MODULO 'CTINST' ATE 'CTPOP'                                           EMU11440
                                             1145  *-----------------------------------------------------------------------EMU11450
          0008                               1146         BND                                                          EMU11460
     2007 0008 4600 1010 1000 0000 0000      1147  ZERACT JDS                  LOAI                                    EMU11470
0E17      0009 0E17 0000 1110 0001 0111      1148         DC                   CTINST                                  EMU11480
A    GR0  000A 2240 0010 0010 0100 0000      1149         MSGR                 A,GR0                                   EMU11490
001F D    000B C07E 1100 0000 0111 1110      1150         MID                  31,D                                    EMU11500
D    B    000C 3182 0011 0001 1000 0010      1151         MSRR                 D,B                                     EMU11510
GR0  A    000D 2004 0010 0000 0000 0100      1152         MSGR                 GR0,A                                   EMU11520
0000 D    000E C002 1100 0000 0000 0010      1153  ZZZZ   MID                  0,D                                     -EMU11520
          000F 0802 0000 1000 0000 0010      1154         CMNT                                                         EMU11530
A    GR0  0010 1240 0001 0010 0100 0000      1155         MURG    +1           A,GR0                                   EMU11540
B    B    0011 116A 0001 0001 0110 1010      1156         MURR    -1           B,B                                     EMU11550
S    0000 0012 9C7A 1001 1100 0111 1010      1157         TST          71      S,ZZZZ                                  -EMU11560
                                                                                                                       EMU11570
```

EDISA ED300 * MONTADOR DE MICROPROGRAMAS * MICRON E24-R000 ***(MICROT)*** PAG 0021 86/11/10

```
ORIG DEST LOCN INST CONFIGURACAO   BITS NIVS CM .    1    .    2    .    3    .    4    .    5    .    6    .    7    .    8
STK  MPC                                      ROTULO OPER S M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO                    ID/SEQ
          0013 3307 0011 0011 0000 0111 1158        MSRH                STK,MPC                                      EMU11580
                                        1159 *-------------------------ESTATI---------------------------------------EMU11590
                                        1160 *----------IMPRESSAO DOS CABECALHOS:                                    EMU11600
          0014                          1161        PNO                                                              EMU11610
     237C 0014 A80F 1010 1000 1101 1111 1162 ESTATI JDS                 CLEAR                                        EMU11620
2521      0015 2521 0010 0101 0010 0001 1163        DC                  ASI                                          EMU11630
0088      0016 0088 0000 0000 1000 1000 1164        DC                  136                                          EMU11640
     2000 0017 A800 1010 1000 0000 0000 1165        JDS                 LDAI                                         EMU11650
25A9      0018 25A9 0010 0101 1010 1001 1166        DC                  ACI                                          EMU11660
0001 0    0019 C006 1100 0000 0000 0110 1167        MID                 1,0                                          EMU11670
          001A 0802 0000 1000 0000 0010 1168        CMWT                                                             EMU11680
     2000 001B A800 1010 1000 0000 0000 1169        JDS                 LDAI                                         EMU11690
2521      001C 2521 0010 0101 0010 0001 1170        DC                  ASI                                          EMU11700
A    B    001D 3232 0011 0010 0000 0010 1171        MSRH                A,B                                          EMU11710
0030 0    001E C0C2 1100 0000 1100 0010 1172        MID                 48,0                                         EMU11720
     GR2  001F 1202 0001 0010 0000 0010 1173        MAG     +2          GR2                                          EMU11730
     2028 0020 A80A 1010 1000 0000 1010 1174        JDS                 STKSLV                                       EMU11740
0900      0021 0900 0000 1001 1101 0000 1175        DC                  ICPMEN                                       EMU11750
0075      0022 0D75 0000 1101 0111 0101 1176        DC                  EST1                                         EMU11760
     2514 0023 A945 1010 1001 0100 0101 1177        JDS                 PRINT                                        EMU11770
     2028 0024 A80A 1010 1000 0000 1010 1178        JDS                 STKSLV                                       EMU11780
0900      0025 0900 0000 1001 1101 0000 1179        DC                  ICPMEN                                       EMU11790
0085      0026 0D85 0000 1101 1000 0101 1180        DC                  EST2                                         EMU11800
     2514 0027 A945 1010 1001 0100 0101 1181        JDS                 PRINT                                        EMU11810
                                        1182 *-----IMPRESSAO DOS CONTADORES:                                         EMU11820
                                        1183 *                          MONTA LINHA                                  EMU11830
001F 0    0028 C07E 1100 0000 0111 1110 1184        MID                 31,0                                         EMU11840
0    GR0  0029 21C0 0010 0001 1100 0000 1185        MSRG                0,GR0          GR0=CONTADOR DE LINHAS        EMU11850
     2000 002A A800 1010 1000 0000 0000 1186        JDS                 LDAI                                         EMU11860
0E17      002B 0E17 0000 1110 0001 0111 1187        DC                  CTINST         AK<--CTINST 'PRIM. CONTADOR'  EMU11870
A    GR1  002C 2241 0010 0010 0100 0001 1188        MSRG                A,GR1          GR1=ENDERECO DO CONTADOR      EMU11880
     2000 002D A800 1010 1000 0000 0000 1189        JDS                 LDAI                                         EMU11890
0D97      002E 0D97 0000 1101 1001 0111 1190        DC                  CEST1                                        EMU11900
A    GR4  002F 2244 0010 0010 0100 0100 1191        MSRG                A,GR4          GR4=ENDERECO DADOS IMPRESS.   EMU11910
GR1  A    0030 2084 0010 0000 1000 0100 1192 EMUINI MSGF                GR1,A                                        EMU11920
          0031 0803 0000 1000 0000 0011 1193        CMRD                                                             EMU11930
0    GR6  0032 21C6 0010 0001 1100 0110 1194        MSRG                0,GR6          DK<--CONTEUDO DO CONTADOR     EMU11940
     2000 0033 A800 1010 1000 0000 0000 1195        JDS                 LDAI                                         EMU11950
0071      0034 0D71 0000 1101 0111 0001 1196        DC                  STHEXA                                       EMU11960
     2023 0035 A80A 1010 1000 0000 1010 1197        JDS                 STKSLV                                       EMU11970
2294      0036 2294 0010 0010 1001 0100 1198        DC                  3H                                           EMU11980
     005C 0037 4357 1010 0011 0101 0111 1199        JDS                 FAZSTR                                       EMU11990
     2003 0038 A802 1010 1000 0000 0010 1200        JDS                 SDD                                          EMU12000
006F      0039 0D6F 0000 1101 0110 1111 1201        DC                  LINCT2         2 CARAC. DO STRING            EMU12010
     005C 003A 4357 1010 0011 0101 0111 1202        JDS                 FAZSTR                                       EMU12020
     2003 003B A802 1010 1000 0000 0010 1203        JDS                 SDD                                          EMU12030
006E      003C 0D6E 0000 1101 0110 1110 1204        DC                  LINCT1         2 CARACTERES DO STRING        EMU12040
0004 0    003D C012 1100 0000 0001 0010 1205        MID                 4,0                                          EMU12050
0    GR5  003E 21C5 0010 0001 1100 0101 1206        MSRG                0,GR5          GR5=CONTADOR DOS 'CEST'       EMU12060
     2000 003F A800 1010 1000 0000 0000 1207        JDS                 LDAI                                         EMU12070
0D64      0040 0D64 0000 1101 0110 0100 1208        DC                  LINHAE                                       EMU12080
A    GR3  0041 2243 0010 0010 0100 0011 1209        MSRG                A,GR3          GR3=ENDERECO LINHA MONTAGEMEMU12090
GR4  A    0042 2204 0010 0010 0000 0100 1210 EMUVOL MSGF                GR4,A                                        EMU12100
          0043 0803 0000 1000 0000 0011 1211        CMRD                                                             EMU12110
A    GR4  0044 1244 0001 0010 0100 0100 1212        MURG    +1          A,GR4                                        EMU12120
GR3  A    0045 2184 0010 0001 1000 0100 1213        MSCR                GR3,A                                        EMU12130
          0046 0802 0000 1000 0000 0010 1214        CMWT                                                             EMU12140
A    GR3  0047 1243 0001 0010 0100 0011 1215        MURG    +1          A,GR3                                        EMU12150
```

```
ORIG  DEST  LOCC  INST  CONFIGURACAO   BITS  NIMS  C . 1 . 2 . 3 . 4 . 5 . 6 . 7 . 8
                                                  ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                    ID/SEQ
GR5   B     0046  2262  0010 0010 1000 0010  1216        MSGR              GR5,B                                          EMU12160
B     GR6   0049  1140  0001 0001 0100 1101  1217        MVRG   -1         B,GR5                                          EMU12170
S     0042  004A  9C77  1001 1100 0111 0111  1218        TST            71 S,EMUVOL                                       EMU12180
                        1219 *------IMPRIME A LINHA MONTADA:                                                             EMU12190
      2000  004B  4800  1010 1000 0000 0000  1220        JDS               LDAI                                           EMU12200
2521        004C  2521  0010 0101 0010 0001  1221        DC                ASI                                            EMU12210
A     B     004D  3202  0011 0010 0000 0010  1222        MSRB              A,B                                            EMU12220
0030  D     004E  C0C2  1100 0000 1100 0010  1223        MID               48,D                                           EMU12230
      GR2   004F  1202  0001 0010 0000 0010  1224        MAR    +0         GR2                                            EMU12240
      2028  0050  A80A  1010 1000 0000 1010  1225        JDS               STKSLV                                         EMU12250
0900        0051  0900  0000 1001 1101 0000  1226        DC                ICPMEN                                         EMU12260
0064        0052  0064  0000 1101 0110 0100  1227        DC                LINHAE                                         EMU12270
      2514  0053  A945  1010 1001 0100 0101  1228        JDS               PRINT                                          EMU12280
GR1   B     0054  2082  0010 0000 1000 0010  1229        MSGR              GR1,B                                          EMU12290
B     GR1   0055  1141  0001 0001 0100 0001  1230        MVRG   +1         B,GR1     PROXIMO CONTADOR                     EMU12300
GR0   B     0056  2002  0010 0000 0000 0010  1231        MSGR              GR0,B                                          EMU12310
B     GR0   0057  1148  0001 0001 0100 1000  1232        MVRG   -1         B,GR0     CONTLINHA=CONTLINHA-1                EMU12320
S     0054  0058  9071  1001 0000 0111 0001  1233        TST            70 S,**2                                          EMU12330
      0030  0059  0C63  0000 0110 0011 0011  1234        JMU               EMUINI                                         EMU12340
STK   MPC   005A  3307  0011 0011 0000 0111  1235        MSRB              STK,MPC   VOLTA P/ GERENCIADOR                  EMU12350
                        1236 *----------------------SUBROTINA 'FAZSTR'----------------------------------------------------EMU12360
                  005C                      1237        END                                                              EMU12370
A     A     005C  126C  0001 0010 0110 1100  1238 FAZSTR MVRR   -1         A,A                                            EMU12380
                  005D  0803  0000 1000 0000 0011  1239        CMRD                                                       EMU12390
D     B     005E  3152  0011 0001 1001 0010  1240        MSRR              D,B                                            EMU12400
A     A     005F  126C  0001 0010 0110 1100  1241        MVRR   -1         A,A                                            EMU12410
                  0060  0803  0000 1000 0000 0011  1242        CMRD                                                       EMU12420
D     D     0061  3186  0011 0001 1010 1011  1243        MSRR   X          D,D                                            EMU12430
      D     0062  1223  0001 0010 0010 0011  1244        MAR    +0         D                                              EMU12440
STK   MPC   0063  3307  0011 0011 0000 0111  1245        MSRB              STK,MPC                                        EMU12450
                        1246 *LINHA DE MONTAGEM PARA 'ICPMEN' PREENCHER A 'ASI':                                        EMU12460
                  0064                      1247 LINHAE DA                 4                                              EMU12470
2020        0065  2020  0010 0000 0010 0000  1248        DC                X'2020'                                        EMU12480
2020        0066  2020  0010 0000 0010 0000  1249        DC                X'2020'                                        EMU12490
2020        0067  2020  0010 0000 0010 0000  1250        DC                X'2020'                                        EMU12500
2020        0068  2020  0010 0000 0010 0000  1251        DC                X'2020'                                        EMU12510
2020        0069  2020  0010 0000 0010 0000  1252        DC                X'2020'                                        EMU12520
2020        006A  2020  0010 0000 0010 0000  1253        DC                X'2020'                                        EMU12530
                  006B                      1254 LINCT1 DA                 1                                              EMU12540
                  006F                      1255 LINCT2 DA                 1                                              EMU12550
200A        0070  200A  0010 0000 0000 1010  1256        DC                X'200A'                                        EMU12560
                        1257 *FORMACAO DO STRING USADO NA ROTINA 'SH'                                                   EMU12570
                  0071                      1258 STHEXA DA                 4                                              EMU12580
                        1259 *----------------------DADOS PARA IMPRESSAO--------------------------------------------------EMU12590
4553        0075  4553  0100 0101 0101 0011  1260 EST1   DC                X'4553'   ES                                   EMU12600
5441        0076  5441  0101 0100 0100 0001  1261        DC                X'5441'   TA                                   EMU12610
5449        0077  5449  0101 0100 0100 1001  1262        DC                X'5449'   TI                                   EMU12620
5354        0078  5354  0101 0011 0101 0100  1263        DC                X'5354'   ST                                   EMU12630
4943        0079  4943  0100 1001 0100 0011  1264        DC                X'4943'   IC                                   EMU12640
4120        007A  4120  0100 0001 0010 0000  1265        DC                X'4120'   A                                    EMU12650
4441        007B  4441  0100 0100 0100 0001  1266        DC                X'4441'   DA                                   EMU12660
5320        007C  5320  0101 0011 0010 0000  1267        DC                X'5320'   S                                    EMU12670
494E        007D  494E  0100 1001 0100 1110  1268        DC                X'494E'   IN                                   EMU12680
5354        007E  5354  0101 0011 0101 0100  1269        DC                X'5354'   ST                                   EMU12690
5255        007F  5255  0101 0010 0101 0101  1270        DC                X'5255'   RU                                   EMU12700
434F        0080  434F  0100 0011 0100 1111  1271        DC                X'434F'   CO                                   EMU12710
4553        0081  4553  0100 0101 0101 0011  1272        DC                X'4553'   ES                                   EMU12720
2050        0082  2050  0010 0000 0101 0000  1273        DC                X'2050'   P                                    EMU12730
```

```
                                                        C   .   1   .   2   .   3   .   4   .   5   .   6   .   7   .   8
ORIG  DEST  LOCN  INST  CONFIGURACAO         BITS  NINS  ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO  COMENTARIO                ID/SEQ
4349        0083  4349  0100 0011 0100 1001        1274         DC                  X'4349'         CI                        EMU12740
520A        0084  520A  0101 0010 0000 1010        1275         DC                  X'520A'         R.                        EMU12750
494E        0085  494E  0100 1001 0100 1110        1276  EST2   DC                  X'494E'         IN                        EMU12760
5354        0086  5354  0101 0011 0101 0100        1277         DC                  X'5354'         ST                        EMU12770
5255        0087  5255  0101 0010 0101 0101        1278         DC                  X'5255'         RU                        EMU12780
4341        0088  4341  0100 0011 0100 0001        1279         DC                  X'4341'         CA                        EMU12790
4F20        0089  4F20  0100 1111 0010 0000        1280         DC                  X'4F20'         O                         EMU12800
2020        008A  2020  0010 0000 0010 0000        1281         DC                  X'2020'                                   EMU12810
2020        008B  2020  0010 0000 0010 0000        1282         DC                  X'2020'                                   EMU12820
2020        008C  2020  0010 0000 0010 0000        1283         DC                  X'2020'                                   EMU12830
2020        008D  2020  0010 0000 0010 0000        1284         DC                  X'2020'                                   EMU12840
2020        008E  2020  0010 0000 0010 0000        1285         DC                  X'2020'                                   EMU12850
5155        008F  5155  0101 0001 0101 0101        1286         DC                  X'5155'         QU                        EMU12860
414E        0090  414E  0100 0001 0100 1110        1287         DC                  X'414E'         AN                        EMU12870
5449        0091  5449  0101 0100 0100 1001        1288         DC                  X'5449'         TI                        EMU12880
4441        0092  4441  0100 0100 0100 0001        1289         DC                  X'4441'         DA                        EMU12890
4445        0093  4445  0100 0100 0100 0101        1290         DC                  X'4445'                                   EMU12900
2D68        0094  2D68  0010 1101 0110 1000        1291         DC                  X'2D68'         --                        EMU12910
6578        0095  6578  0110 0101 0111 1000        1292         DC                  X'6578'         EX                        EMU12920
610A        0096  610A  0110 0001 0000 1010        1293         DC                  X'610A'         A                         EMU12930
                                                   1294  *------------INSTRUCOES OP-------------------------------------------  EMU12940
494E        0097  494E  0100 1001 0100 1110        1295  CEST1  DC                  X'494E'                                   EMU12950
5354        0098  5354  0101 0011 0101 0100        1296         DC                  X'5354'                                   EMU12960
2E20        0099  2E20  0010 1110 0010 0000        1297         DC                  X'2E20'                                   EMU12970
2020        009A  2020  0010 0000 0010 0000        1298         DC                  X'2020'                                   EMU12980
494E        009B  494E  0100 1001 0100 1110        1299  CEST2  DC                  X'494E'                                   EMU12990
5354        009C  5354  0101 0011 0101 0100        1300         DC                  X'5354'                                   EMU13000
2E49        009D  2E49  0010 1110 0100 1001        1301         DC                  X'2E49'                                   EMU13010
4D20        009E  4D20  0100 1101 0010 0000        1302         DC                  X'4D20'                                   EMU13020
494E        009F  494E  0100 1001 0100 1110        1303  CEST3  DC                  X'494E'                                   EMU13030
5354        00A0  5354  0101 0011 0101 0100        1304         DC                  X'5354'                                   EMU13040
2E4E        00A1  2E4E  0010 1110 0100 1110        1305         DC                  X'2E4E'                                   EMU13050
494D        00A2  494D  0100 1001 0100 1101        1306         DC                  X'494D'                                   EMU13060
4F50        00A3  4F50  0100 1111 0101 0000        1307  CEST4  DC                  X'4F50'                                   EMU13070
2020        00A4  2020  0010 0000 0010 0000        1308         DC                  X'2020'                                   EMU13080
2020        00A5  2020  0010 0000 0010 0000        1309         DC                  X'2020'                                   EMU13090
2020        00A6  2020  0010 0000 0010 0000        1310         DC                  X'2020'                                   EMU13100
2020        00A7  2020  0010 0000 0010 0000        1311  CEST5  DC                  X'2020'                                   EMU13110
2041        00A8  2041  0010 0000 0100 0001        1312         DC                  X'2041'                                   EMU13120
4444        00A9  4444  0100 0100 0100 0100        1313         DC                  X'4444'                                   EMU13130
2020        00AA  2020  0010 0000 0010 0000        1314         DC                  X'2020'                                   EMU13140
2020        00AB  2020  0010 0000 0010 0000        1315  CEST6  DC                  X'2020'                                   EMU13150
2041        00AC  2041  0010 0000 0100 0001        1316         DC                  X'2041'                                   EMU13160
4444        00AD  4444  0100 0100 0100 0100        1317         DC                  X'4444'                                   EMU13170
4320        00AE  4320  0100 0011 0010 0000        1318         DC                  X'4320'                                   EMU13180
2020        00AF  2020  0010 0000 0010 0000        1319  CEST7  DC                  X'2020'                                   EMU13190
2053        00B0  2053  0010 0000 0101 0011        1320         DC                  X'2053'                                   EMU13200
5542        00B1  5542  0101 0101 0100 0010        1321         DC                  X'5542'                                   EMU13210
2020        00B2  2020  0010 0000 0010 0000        1322         DC                  X'2020'                                   EMU13220
2020        00B3  2020  0010 0000 0010 0000        1323  CEST8  DC                  X'2020'                                   EMU13230
2053        00B4  2053  0010 0000 0101 0011        1324         DC                  X'2053'                                   EMU13240
4243        00B5  4243  0100 0010 0100 0011        1325         DC                  X'4243'                                   EMU13250
2020        00B6  2020  0010 0000 0010 0000        1326         DC                  X'2020'                                   EMU13260
2020        00B7  2020  0010 0000 0010 0000        1327  CEST9  DC                  X'2020'                                   EMU13270
2041        00B8  2041  0010 0000 0100 0001        1328         DC                  X'2041'                                   EMU13280
4449        00B9  4449  0100 0100 0100 1001        1329         DC                  X'4449'                                   EMU13290
2020        00BA  2020  0010 0000 0010 0000        1330         DC                  X'2020'                                   EMU13300
2020        00BB  2020  0010 0000 0010 0000        1331  CEST10 DC                  X'2020'                                   EMU13310
```

EDISA EQ300 * MONTADOR DE MICROPROGRAMAS * MICHON E24-R000 ***(MICROT)*** PAG 0024 86/11/10

ORIG	DEST	LOCN	INST	CONFIGURACAO	BITS	NINS	ROTULO	OPER	M1/M2	C1	C2	ORIGEM/DESTINO	COMENTARIO	ID/SEQ
2041		0DBC	2041	0010 0000 0100 0001	1332		DC					X'2041'		EMU13320
4E44		0DBD	4E44	0100 1110 0100 0100	1333		DC					X'4E44'		EMU13330
2020		0DBE	2020	0010 0000 0010 0000	1334		DC					X'2020'		EMU13340
2020		0DBF	2020	0010 0000 0010 0000	1335	CEST11	DC					X'2020'		EMU13350
204F		0DC0	204F	0010 0000 0100 1111	1336		DC					X'204F'		EMU13360
5220		0DC1	5220	0101 0010 0010 0000	1337		DC					X'5220'		EMU13370
2020		0DC2	2020	0010 0000 0010 0000	1338		DC					X'2020'		EMU13380
2020		0DC3	2020	0010 0000 0010 0000	1339	CEST12	DC					X'2020'		EMU13390
2058		0DC4	2058	0010 0000 0101 1000	1340		DC					X'2058'		EMU13400
4F52		0DC5	4F52	0100 1111 0101 0010	1341		DC					X'4F52'		EMU13410
2020		0DC6	2020	0010 0000 0010 0000	1342		DC					X'2020'		EMU13420
					1343	*----------INSTRUCOES OPK----------								EMU13430
4F50		0DC7	4F50	0100 1111 0101 0000	1344	CEST13	DC					X'4F50'		EMU13440
4620		0DC8	4E20	0100 1011 0010 0000	1345		DC					X'4620'		EMU13450
2020		0DC9	2020	0010 0000 0010 0000	1346		DC					X'2020'		EMU13460
2020		0DCA	2020	0010 0000 0010 0000	1347		DC					X'2020'		EMU13470
2020		0DCB	2020	0010 0000 0010 0000	1348	CEST14	DC					X'2020'		EMU13480
2048		0DCC	2048	0010 0000 0100 1011	1349		DC					X'2048'		EMU13490
3120		0DCD	3120	0011 0001 0010 0000	1350		DC					X'3120'		EMU13500
2020		0DCE	2020	0010 0000 0010 0000	1351		DC					X'2020'		EMU13510
2020		0DCF	2020	0010 0000 0010 0000	1352	CEST15	DC					X'2020'		EMU13520
2048		0DD0	2048	0010 0000 0100 1011	1353		DC					X'2048'		EMU13530
2031		0DD1	2031	0010 1101 0011 0001	1354		DC					X'2031'		EMU13540
2020		0DD2	2020	0010 0000 0010 0000	1355		DC					X'2020'		EMU13550
2020		0DD3	2020	0010 0000 0011 0000	1356	CEST16	DC					X'2020'		EMU13560
2048		0DD4	2048	0010 0000 0100 1011	1357		DC					X'2048'		EMU13570
2031		0DD5	2031	0010 1101 0011 0001	1358		DC					X'2031'		EMU13580
2020		0DD6	2020	0010 0000 0010 0000	1359		DC					X'2020'		EMU13590
2020		0DD7	2020	0010 0000 0010 0000	1360	CEST16	DC					X'2020'		EMU13600
2048		0DD8	2048	0010 0000 0100 1011	1361		DC					X'2048'		EMU13610
3020		0DD9	3020	0011 0000 0010 0000	1362		DC					X'3020'		EMU13620
2020		0DDA	2020	0010 0000 0010 0000	1363		DC					X'2020'		EMU13630
					1364	*----------INSERCAO E EXTRACAO DE BYTES----------								EMU13640
4945		0DDB	4945	0100 1001 0100 0101	1365	CEST17	DC					X'4945'		EMU13650
2020		0DDC	2020	0010 0000 0010 0000	1366		DC					X'2020'		EMU13660
2020		0DDD	2020	0010 0000 0010 0000	1367		DC					X'2020'		EMU13670
2020		0DDE	2020	0010 0000 0010 0000	1368		DC					X'2020'		EMU13680
2020		0DDF	2020	0010 0000 0010 0000	1369	CEST18	DC					X'2020'		EMU13690
2049		0DE0	2049	0010 0000 0100 1001	1370		DC					X'2049'		EMU13700
4E53		0DE1	4E53	0100 1110 0101 0011	1371		DC					X'4E53'		EMU13710
4220		0DE2	4220	0100 0010 0010 0000	1372		DC					X'4220'		EMU13720
2020		0DE3	2020	0010 0000 0010 0000	1373	CEST19	DC					X'2020'		EMU13730
2045		0DE4	2045	0010 0000 0100 0101	1374		DC					X'2045'		EMU13740
5654		0DE5	5654	0101 0110 0101 0100	1375		DC					X'5654'		EMU13750
4220		0DE6	4220	0100 0010 0010 0000	1376		DC					X'4220'		EMU13760
					1377	*----------TESTE E DESVIO----------								EMU13770
5345		0DE7	5345	0101 0011 0100 0101	1378	CEST20	DC					X'5345'		EMU13780
5120		0DE8	5120	0101 0001 0010 0000	1379		DC					X'5120'		EMU13790
2020		0DE9	2020	0010 0000 0010 0000	1380		DC					X'2020'		EMU13800
2020		0DEA	2020	0010 0000 0010 0000	1381		DC					X'2020'		EMU13810
5347		0DEB	5347	0101 0011 0100 0111	1382	CEST21	DC					X'5347'		EMU13820
5420		0DEC	5420	0101 0100 0010 0000	1383		DC					X'5420'		EMU13830
2020		0DED	2020	0010 0000 0010 0000	1384		DC					X'2020'		EMU13840
2020		0DEE	2020	0010 0000 0010 0000	1385		DC					X'2020'		EMU13850
5347		0DEF	5347	0101 0011 0100 0111	1386	CEST22	DC					X'5347'		EMU13860
4520		0DF0	4520	0100 0101 0010 0000	1387		DC					X'4520'		EMU13870
2020		0DF1	2020	0010 0000 0010 0000	1388		DC					X'2020'		EMU13880
2020		0DF2	2020	0010 0000 0010 0000	1389		DC					X'2020'		EMU13890

```
ORIG  DEST  LOCN  INST  CONFIGURACAO  BITS   NUMS  ROTULO OPER & M1/M2  C1 C2 ORIGEM/DESTINO   COMENTARIO                ID/SEQ
                                                   0 .    1    .    2    .    3    .    4    .    5    .    6    .    7    .    8
4246        0DF3  4246  0100 0010 0100 0110  1390  CEST23 DC        X'4246'                                            EMU13900
2020        0DF4  2020  0010 0000 0010 0000  1391         DC        X'2020'                                            EMU13910
2020        0DF5  2020  0010 0000 0010 0000  1392         DC        X'2020'                                            EMU13920
2020        0DF6  2020  0010 0000 0010 0000  1393         DC        X'2020'                                            EMU13930
                                             1394  *----------TRANSFERENCIA ENTRE REGISTRADORES---------------------------EMU13940
5243        0DF7  5243  0101 0010 0100 0011  1395  CEST24 DC        X'5243'                                            EMU13950
2020        0DF8  2020  0010 0000 0010 0000  1396         DC        X'2020'                                            EMU13960
2020        0DF9  2020  0010 0000 0010 0000  1397         DC        X'2020'                                            EMU13970
2020        0DFA  2020  0010 0000 0010 0000  1398         DC        X'2020'                                            EMU13980
2020        0DFB  2020  0010 0000 0010 0000  1399  CEST25 DC        X'2020'                                            EMU13990
2052        0DFC  2052  0010 0000 0101 0010  1400         DC        X'2052'                                            EMU14000
4F4C        0DFD  4F4C  0100 1111 0100 1100  1401         DC        X'4F4C'                                            EMU14010
4320        0DFE  4320  0100 0011 0010 0000  1402         DC        X'4320'                                            EMU14020
2020        0DFF  2020  0010 0000 0010 0000  1403  CEST26 DC        X'2020'                                            EMU14030
2052        0E00  2052  0010 0000 0101 0010  1404         DC        X'2052'                                            EMU14040
4F52        0E01  4F52  0100 1111 0101 0010  1405         DC        X'4F52'                                            EMU14050
4320        0E02  4320  0100 0011 0010 0000  1406         DC        X'4320'                                            EMU14060
2020        0E03  2020  0010 0000 0010 0000  1407  CEST27 DC        X'2020'                                            EMU14070
2040        0E04  2040  0010 0000 0100 0000  1408         DC        X'2040'                                            EMU14080
4F56        0E05  4F56  0100 1111 0101 0110  1409         DC        X'4F56'                                            EMU14090
4520        0E06  4520  0100 0101 0010 0000  1410         DC        X'4520'                                            EMU14100
                                             1411  *----------INSTRUCOES QUE ACESSAM MEMORIA------------------------------EMU14110
5354        0E07  5354  0101 0011 0101 0100  1412  CEST28 DC        X'5354'                                            EMU14120
4F52        0E08  4F52  0100 1111 0101 0010  1413         DC        X'4F52'                                            EMU14130
4520        0E09  4520  0100 0101 0010 0000  1414         DC        X'4520'                                            EMU14140
2020        0E0A  2020  0010 0000 0010 0000  1415         DC        X'2020'                                            EMU14150
5055        0E0B  5055  0101 0000 0101 0101  1416  CEST29 DC        X'5055'                                            EMU14160
5343        0E0C  5343  0101 0011 0100 0011  1417         DC        X'5343'                                            EMU14170
2020        0E0D  2020  0010 0000 0010 0000  1418         DC        X'2020'                                            EMU14180
2020        0E0E  2020  0010 0000 0010 0000  1419         DC        X'2020'                                            EMU14190
4C4F        0E0F  4C4F  0100 1100 0100 1111  1420  CEST30 DC        X'4C4F'                                            EMU14200
4144        0E10  4144  0100 0001 0100 0100  1421         DC        X'4144'                                            EMU14210
2020        0E11  2020  0010 0000 0010 0000  1422         DC        X'2020'                                            EMU14220
2020        0E12  2020  0010 0000 0010 0000  1423         DC        X'2020'                                            EMU14230
504F        0E13  504F  0101 0000 0100 1111  1424  CEST31 DC        X'504F'                                            EMU14240
5020        0E14  5020  0101 0000 0010 0000  1425         DC        X'5020'                                            EMU14250
2020        0E15  2020  0010 0000 0010 0000  1426         DC        X'2020'                                            EMU14260
2020        0E16  2020  0010 0000 0010 0000  1427         DC        X'2020'                                            EMU14270
                                             1428  *---FIM DA DEFINICAO DOS DADOS PARA IMPRESSAO--------------------------EMU14280
                                             1429  *--------------------------CONTADORES----------------------------------EMU14290
            0E17                             1430  CTINST DA               1        CONTA INSTRUCOES                  EMU14300
            0E18                             1431  CTIM   DA               1        CONTA INST. C/IMEDIATO            EMU14310
            0E19                             1432  CTNIM  DA               1        CONTA INST. S/IMEDIATO            EMU14320
                                             1433  *----------CONTADORES DAS INSTRUCOES OP--------------------------------EMU14330
            0E1A                             1434  CTOP   DA               1                                          EMU14340
            0E1B                             1435  CTADD  DA               1                                          EMU14350
            0E1C                             1436  CTADDC DA               1                                          EMU14360
            0E1D                             1437  CTSUB  DA               1                                          EMU14370
            0E1E                             1438  CTSBC  DA               1                                          EMU14380
            0E1F                             1439  CTADI  DA               1                                          EMU14390
            0E20                             1440  CTAND  DA               1                                          EMU14400
            0E21                             1441  CTOR   DA               1                                          EMU14410
            0E22                             1442  CTXOR  DA               1                                          EMU14420
                                             1443  *---------CONTADORES DAS INSTRUCOES OPK--------------------------------EMU14430
            0E23                             1444  CTOPK  DA               1                                          EMU14440
            0E24                             1445  CTKUM  DA               1        CONSTANTE 1                       EMU14450
            0E25                             1446  CTKUMN DA               1        CONSTANTE -1                      EMU14460
            0E26                             1447  CTKZER DA               1        CONSTANTE ZERO                    EMU14470
```

```
ORIG DEST LOCN INST CONFIGURACAO  BITS NINS   C    .    1    .    2    .    3    .    4    .    5    .    6    .    7    .    8
                                             ROTULO OPER & M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO                          ID/SEQ
                                        1448 *---------CONTADORES DAS INST. DE INSERCAO E EXTRACAO DE BYTES-------------EMU14480
          0E27                          1449  CTIE    DA                                                               EMU14490
          0E28                          1450  CTINSR  DA                                                               EMU14500
          0E29                          1451  CTEXTR  DA                                                               EMU14510
                                        1452 *----------------CONTADORES DAS INSTRUCOES DE TESTE E DESVIO--------------EMU14520
          0E2A                          1453  CTSEL   DA                                                               EMU14530
          0E2B                          1454  CTTST   DA                                                               EMU14540
          0E2C                          1455  CTSSB   DA                                                               EMU14550
          0E2D                          1456  CTIF    DA                                                               EMU14560
                                        1457 *-----------CONTADORES DAS INSTRUCOES DE TRANSF. ENTRE REGISTRADORES------EMU14570
          0E2E                          1458  CTRC    DA                                                               EMU14580
          0E2F                          1459  CTROLC  DA                                                               EMU14590
          0E30                          1460  CTRORC  DA                                                               EMU14600
          0E31                          1461  CTMOVE  DA                                                               EMU14610
                                        1462 *--------CONTADORES DAS INSTRUCOES QUE ACESSAM MEMORIA--------------------EMU14620
          0E32                          1463  CTSTOR  DA                                                               EMU14630
          0E33                          1464  CTPUSH  DA                                                               EMU14640
          0E34                          1465  CTLOAD  DA                                                               EMU14650
          0E35                          1466  CTPOP   DA                                                               EMU14660
                                        1467 *-----------------------FIM DOS CONTADORES---------------------------------EMU14670
                                        1468 *----------------------------TRLPC---------------------------------------EMU14680
                                        1469 *ESTA ROTINA TRANSFERE MEMORIA PRINCIPAL PARA MEMORIA DE CONTROLE           EMU14690
                                        1470 *GR1=ENDERECO INICIAL DA MEMORIA PRINCIPAL                                 EMU14700
                                        1471 *GR0=ENDERECO INICIAL DA MEMORIA DE CONTROLE                               EMU14710
                                        1472 *R=QUANTIDADE DE PALAVRAS A TRANSFERIR                                     EMU14720
                                        1473 *OBS: ANTES DE CHAMAR A ROTINA, ZERAR BIT 14 DO REGISTRADOR S              EMU14730
                                        1474 *----------------------------------------------------------------------EMU14740
          0E37                          1475          RND                                                              EMU14750
GR1  A    0E37 [illegible]              1476  TRLPC   MSTR  F          GR1,A                                            EMU14760
          [illegible]                   1477-1484  [illegible]  ...  71  D,TRLPC  STK,MPC                            EMU14770-EMU14840
                                        1485 *------------------------------IMAVID---------------------------------EMU14850
                                        1486 *ESTA ROTINA IMPRIME MENSAGEM NO VIDEO E ESPERA RESPOSTA. O ENDERECO DA   EMU14860
                                        1487 *MENSAGEM E PASSADO PELA PALAVRA SEGUINTE A INSTRUCAO DE CHAMADA           EMU14870
                                        1488 *----------------------------------------------------------------------EMU14880
          0E44                          1489          RND                                                              EMU14890
     2000 0E44 A800 1010 1000 0000 0000 1490  IMAVID  JOS              LDAI                                             EMU14900
2776      0E45 2776 0010 0111 0111 0110 1491          DC               ASV                                              EMU14910
A    0E2  0E46 2242 0010 0010 0100 0010 1492          MSRR             A,0E2                                            EMU14920
STK  A    0E47 3304 0011 0011 0000 0100 1493          MSRR             STK,A                                            EMU14930
A         0E48 1267 0001 0010 0110 0111 1494          MJRR    +1       A,D                                              EMU14940
D    STK  0E49 3126 0011 0001 0010 0110 1495          MSRR             D,STK                                            EMU14950
          0E4A 0303 0000 0011 0000 0011 1496          CMRL                                                              EMU14960
D    0E5  0E4B 2105 0010 0001 0000 0101 1497          MSRR             D,0E5                                            EMU14970
     2026 0E4C A80A 1010 1000 0000 1010 1498          JOS              STKSLV                                           EMU14980
0E50      0E4D 0E50 0000 1110 0101 0000 1499          DC               ICINVG                                           EMU14990
STK  MPC  0E4E 3307 0011 0011 0000 0111 1500          MSRR             STK,MPC         'CG' OU 'EX' PODE PROSSEG.      EMU15000
                                        1501 *-----------------------------ICINVG--------------------------------------EMU15010
          0E50                          1502          RND                                                              EMU15020
     2000 0E50 A800 1010 1000 0000 0000 1503  ICINVG  JOS              LDAI                                             EMU15030
2776      0E51 2776 0010 0111 0111 0110 1504          DC               ASV                                              EMU15040
          0E52 C051 1100 0000 0101 0001 1505          MIC              20,[illegible]                                   EMU15050
```

```
                                                           C  .  1  .  2  .  3  .  4  .  5  .  6  .  7  .  8
ORIG DEST LOCN INST CONFIGURACAO        BITS NINS  ROTULO OPER S M1/M2 C1 C2 ORIGEM/DESTINO  COMENTARIO          ID/SEQ
A    0    0E53 3203 0011 0010 0000 0011      1506         MSRR              A,0                                  EMU15060
     GR3  0E54 1203 0001 0010 0000 0011      1507         MAG   +0          GR3                                  EMU15070
     2023 0E55 A80A 1010 1000 0000 1010      1508         JOS               STKSLV                               EMU15080
090A      0E56 090A 0000 1001 1101 1010      1509         DC                ICPLAC                               EMU15090
     263C 0E57 A93F 1010 1001 1000 1111      1510         JOS               OUTIL                                EMU15100
0015      0E58 0015 0000 0000 0001 0101      1511         DC                21                                   EMU15110
     2654 0E59 A995 1010 1001 1001 0101      1512         JOS               INAEV                                EMU15120
     2000 0E5A A803 1010 1000 0000 0000      1513         JOS               LDAI                                 EMU15130
274E      0E5B 274E 0010 0111 0100 1110      1514         DC                AEV                                  EMU15140
0014 6    0E5C C051 1100 0000 0101 0001      1515         MID               20,6                                 EMU15150
A    0    0E5D 3203 0011 0010 0000 0011      1516         MSRR              A,0                                  EMU15160
     A    0E5E 1224 0001 0010 0010 0100      1517         MAR   +0          A                                    EMU15170
     204C 0E5F B813 1011 1000 0001 0011      1518         JOU               STKREC                               EMU15180
                                             1519  *---MENSAGENS EMITIDAS PELO GERENCIADOR NO VIDEO---------------EMU15190
          0E60                               1520         END                                                    EMU15200
4341      0E60 4341 0100 0011 0100 0001      1521  MENSA1 DC                X'4341'         CA                   EMU15210
5247      0E61 5247 0101 0010 0100 0111      1522         DC                X'5247'         RG                   EMU15220
413F      0E62 413F 0100 0001 0011 1111      1523         DC                X'413F'         A?                   EMU15230
0A00      0E63 0A00 0000 1010 0000 0000      1524         DC                X'0A00'         NL                   EMU15240
          0E64                               1525         END                                                    EMU15250
4341      0E64 4341 0100 0011 0100 0001      1526  MENSA2 DC                X'4341'         CA                   EMU15260
5250      0E65 5250 0101 0010 0101 0000      1527         DC                X'5250'         RP                   EMU15270
5420      0E66 5420 0101 0100 0010 0000      1528         DC                X'5420'         T                    EMU15280
0A00      0E67 0A00 0000 1010 0000 0000      1529         DC                X'0A00'         NL                   EMU15290
          0E68                               1530         END                                                    EMU15300
4558      0E68 4558 0100 0101 0101 1000      1531  MENSA3 DC                X'4558'         EX                   EMU15310
3F20      0E69 3F20 0011 1111 0010 0000      1532         DC                X'3F20'         ?                    EMU15320
0A00      0E6A 0A00 0000 1010 0000 0000      1533         DC                X'0A00'         NL                   EMU15330
          0E6B                               1534         END                                                    EMU15340
4558      0E6B 4558 0100 0101 0101 1000      1535  MENSA4 DC                X'4558'         EX                   EMU15350
5054      0E6C 5054 0101 0000 0101 0100      1536         DC                X'5054'         PT                   EMU15360
0A00      0E6D 0A00 0000 1010 0000 0000      1537         DC                X'0A00'         NL                   EMU15370
          0E6E                               1538         END                                                    EMU15380
```

BIBLIOGRAFIA

/ALV 85/ ALVES, Vera Maria dos Santos; BORGES, Elenara; COSTA, Antônio Carlos da Rocha; TOSCANI, Simão Sirineo. Emulador e Software Básico para o Microprocessador Integrado de 16 Bits da UFRGS. In: CONGRESSO NACIONAL DE INFORMÁTICA, 18; São Paulo, set 23-29, 1985. Anais. São Paulo, SUCESU, 1985. p.959-967.

/BAR 72/ BARRON, D.W. Assemblers and loaders 2.ed. New York, American Elswier, 1972.

/BOR 87/ BORGES, Elenara Maria da Silva. Montador cruzado e depurador para o micro PCIR. Porto Alegre, PGCC-UFRGS, 1987. (Dissertação de Mestrado)

/CAL 79/ CALINGAERT, Peter. Assemblers, compilers and program translation. Maryland, Computer Science Press, 1979.

/COR 82/ CORSO, T.B. ; WEBER, T.S.; COSTA, A.C.R. & TODESCO, A.R.W. Suporte para desenvolvimento de firmware no computador ED 300. In: CONGRESSO NACIONAL DE INFORMÁTICA, 15; Rio de Janeiro, out 1982. Anais. Rio de Janeiro, SUCESU, 1982. p.267-271.

/EDI 81/ EDISA. Micmon - Micromontador. Gravataí 1981 (publicação interna)

/FIS 79/ FISCHER, W. P. Microprocessor assembly language draft standard. Computer, New York, 12(12): 96-109, Dez. 1979.

/KNU 73/ KNUTH, D.E. The Art of Computer Programming, Searching and Sorting. Reading Addson Wesley, 1973. V.3

/PAD 80/ PADILHA, Antônio C.M. Características de microprogramação dos computadores da série ED-300. Porto Alegre, PGCC-UFRGS, 1980.

/POE 74/ POEL, W.L. Vander & Maarssen, L.A., ed., Machines oriented higher level languages. In: IFIP WORKING CONFERENCE, Trodhein, Aug. 27-31, 1973. Proceedings. Amsterdam, North-Holland p. c1974.

/ROS 69/ ROSIN, R.F. Comtemporary concepts of microprogramming and emulation. Computing Surveys, New York, 1(4):197-212, Dec. 1969.

/TOD 86/ TODESCO, Antônio R.W. Concepção de um circuito integrado do tipo processador com conjunto de instruções reduzido. Porto Alegre, PGCC-UFRGS, 1986. (Dissertaça~o de Mestrado)

/TOR 82/ TORNQUIST, Martin. Uma linguagem para implementação de sistemas multiprogramados em microcomputadores. Porto Alegre, PGCC-UFRGS, 1982.

/WIR 77/ WIRT, Niklaus. What can we do about the unnecessary diversity of notation for syntatic definitions. Comm. ACM, New York, 20 (11): 822-823, Nov. 1977

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Pós-Graduação em Ciência da Computação

Emulador e montador para o
micro PCIR

Dissertação apresentada aos Srs.

Visto e permitida a impressão
Porto Alegre, 31../.05../88.

Coordenador do Curso de Pós-Graduação
em Ciência da Computação