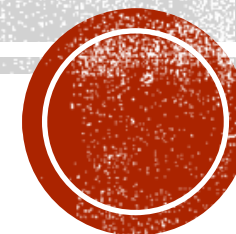


SISTEMAS MICROPROCESSADOS I

Prof.: João Castelo



CENTRO UNIVERSITÁRIO
SENAI CIMATEC



FORMATOS DE INSTRUÇÕES

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Double Operand	Op-Code				S-Reg				Ad	B/W	As		D-Reg			

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Single Operand	Op-Code									B/W	Ad		D/S - Reg			

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Jump	Op-Code			C(ondition)			10-Bit (Two's Complement) PC Offset									

As – The addressing bits responsible for the addressing mode for src

S-reg – The working register used for the source (src)

Ad – The addressing bits responsible for the addressing mode for dst

D-reg – The working register used for the destination (dst)

B/W – Byte or Word: 0-Word , 1-Byte

src – The source operand defined by As and S-reg

dst – The destination operand defined by Ad and D-reg

INSTRUÇÕES COM DOIS OPERANDOS

Instruction	Operation	Description
add src, dst	$\text{src} + \text{dst} \rightarrow \text{dst}$	Add source to destination
addc src, dst	$\text{src} + \text{dst} + C \rightarrow \text{dst}$	Add source and carry to destination
dadd src, dst	$\text{src} + \text{dst} \rightarrow \text{dst}$ (decimal)	Decimal add source and carry to destination
sub src, dst	$\text{dst} + \text{.not src} + 1 \rightarrow \text{dst}$	Subtract source from destination
subc src, dst	$\text{dst} + \text{.not src} + C \rightarrow \text{dst}$	Subtract source and not carry from destination
and src, dst	$\text{src} \text{ .and } \text{dst} \rightarrow \text{dst}$	And source with destination
bic src, dst	$\text{.not src} \text{ .and } \text{dst} \rightarrow \text{dst}$	Clear bits in destination
bis src, dst	$\text{src} \text{ .or } \text{dst} \rightarrow \text{dst}$	Set bits in destination
bit src, dst	$\text{src} \text{ .and } \text{.dst}$	Test bits in destination
xor src, dst	$\text{src} \text{ .xor } \text{dst} \rightarrow \text{dst}$	Xor source with destination
cmp src, dst	$\text{dst} - \text{src}$	Compare source to destination
mov src, dst	$\text{src} \rightarrow \text{dst}$	Copy source to destination

INSTRUÇÕES COM UM OPERANDO

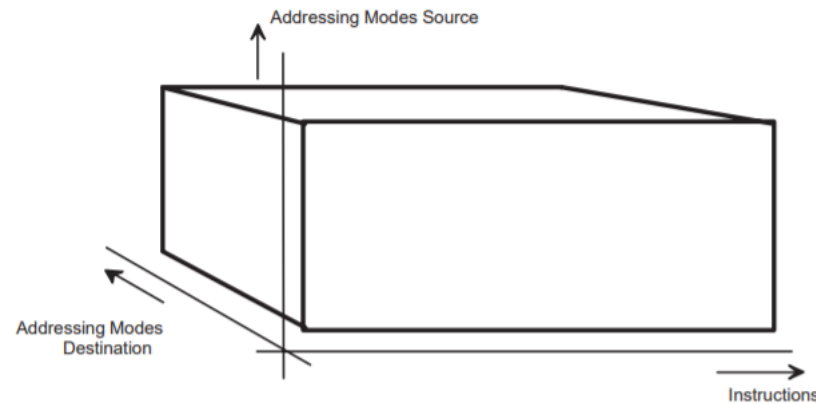
Instruction	Operation	Description
rra dst	MSB \rightarrow MSB \rightarrow ... LSB \rightarrow C	Roll destination right
rrc dst	C \rightarrow MSB \rightarrow ... \rightarrow LSB \rightarrow C	Roll destination right through carry
swpb dst	Swap bytes	Swap bytes in destination
sxt dst	bit7 \rightarrow bit8 \rightarrow ... bit15	Sign extend destination
push dst	SP - 2 \rightarrow SP, src \rightarrow @SP	Push source on stack
call dst	SP - 2 \rightarrow SP, PC + 2 \rightarrow @SP, dst \rightarrow PC	Subroutine call to destination
reti	@SP+ \rightarrow SR, @SP+ \rightarrow @SP	Return from Interrupt

INSTRUÇÃO DE SALTO

Instruction	Condition	Description
jc label	C=1	Jump to label if Carry bit is set
Jge label	$(N \cdot \text{xor } V) = 0$	Jump to label if greater than or equal
Jl label	$(N \cdot \text{xor } V) = 1$	Jump to label if less than
jmp label	No Condition	Unconditional jump
jn label	N=1	Jump to label if the negative bit is set
Jnc label	C=0	Jump to label if carry bit is reset
Jnz/jne label	Z=0	Jump to label if zero bit is reset / jump if not equal
Jz/jcq label	Z=1	Jump to label if zero bit is set

ORTOGONALIDADE

- Esta notação da ciência da computação significa que uma única instrução de operando pode usar qualquer modo de endereçamento ou que qualquer instrução de operando duplo pode usar qualquer combinação de modos de endereçamento de origem e destino.



MODOS DE ENDEREÇAMENTO

Double Operand Instructions

Mnemonic

Source, Destination

Register
Indexed

Absolute

Symbolic

Immediate

Register indirect

Register indirect autoincrement

Register
Indexed

Absolute

Symbolic

Single Operand Instructions

Mnemonic

Destination

Register
Indexed

Absolute

Symbolic

Immediate

Register indirect

Register indirect autoincrement

MODO REGISTRADOR

- O operando está contido em um dos registradores R0 a R15.
- Este é o modo de endereçamento mais rápido e que requer menos memória.

```
; Add the contents of R7 to the contents of R8  
;  
ADD    R7,R8          ; (R7) + (R8) → (R8)
```


MOD0 INDEXADO

- Permite que se utilize uma constante (endereço absoluto), que somada a um registrador vai compor o endereço do operando fonte ou do operando de destino.
- Pode ser utilizado qualquer registrador como índice , exceto o R0 e o R2.
- Esse modo é útil no acesso a tabelas ou matrizes em que o endereço do primeiro elemento da matriz é somado a um índice (conteúdo do registrador) para formar o endereço do elemento desejado.
- Pode ser utilizado tanto para operados fontes quanto para operandos destino, inclusive simultaneamente.

; Carrega o valor 0xAB no endereço 0x200 mais o índice formado por R9

MOV.B #0xAB, 0x200(R9)

MODO SIMBÓLICO

- É uma variação do modo indexado.
- O registrador de índice é o R0 (PC) - o operando fonte ou de destino é obtido pela soma de uma constante imediata ao conteúdo do PC.

```
; copiar o conteúdo de R8 para o endereço representado pelo símbolo TESTE  
;(localizado no endereço 0x0200 da memória) em uma operação de 8 bits  
  
MOV.B R8, TESTE
```

MODO ABSOLUTO

- É uma segunda variação do modo indexado.
- O endereço absoluto do operando fonte ou destino é informado diretamente como operando da instrução.
- A diferença entre o modo absoluto e o modo indexado é que neste caso o registrador utilizado para indexar o endereço é o R2, atuando como gerador de constantes e gerando a constante 0, ou seja, o endereço será o próprio valor do operando.

```
;copiar o conteúdo do endereço especificado pelo símbolo TESTE (0x0200)  
;para o endereço absoluto 0x208
```

```
MOV.B TESTE,&0x208
```

Usamos o operador & para especificar que é um endereço de memória

MOD0 INDIRETO

- O registrador utilizado contém o endereço do operando. O operando pode estar localizado em qualquer lugar em todo o espaço da memória (64K).

```
; Add the byte addressed by R8 to the contents of R9  
;  
ADD.B @R8,R9      ; ((R8)) + (R9) → (R9)
```

O sufixo .B
sinaliza a
operação de 8 bits

MODO INDIRETO COM AUTO-INCREMENTO

- Esse modo é similar ao anterior, mas com a diferença de que o conteúdo do registrador utilizado como apontador é automaticamente incrementado após sua leitura.
- Nesse modo, somente podem ser utilizados os registradores R1 a R15 – o R0 (PC) não pode ser utilizado.
- A depender do tipo de acesso usado na instrução (byte ou word), o registrador apontador é incrementado em 1 (para operação byte) ou em 2 (para operação word).

```
; Copy the byte operand addressed by R8 to R9  
; and increment the pointer register R8 by one afterwards  
;  
MOV.B    @R8+,R9          ; ((R8)) → (R9), (R8) + 1 → (R8)
```


MODO IMEDIATO

- Caracteriza-se pela presença de uma constante operando como fonte da operação.
- Em Assembly, a constante deve ser precedida do símbolo #.
- Só pode ser utilizado para operandos fonte de uma operação.

```
;para carregar o registrador R5 com o valor 0x1000
```

```
MOV #0x1000,R5
```

```
;escrevendo o valor 0x29 no endereço 0x200 da memória
```

```
MOV.B #0x29,&0x200
```

DÚVIDAS?

