

Organização Básica de computadores e linguagem de montagem

Prof. Edson Borin

2º Semestre de 2015

Conjunto de Instruções e programação do IAS

Conjunto de Instruções do IAS

- O IAS possui um conjunto com **20** instruções
 - Transferência de dados: **6**
 - Aritméticas: **8**
 - Salto incondicional: **2**
 - Salto condicional: **2**
 - Modificação de endereço: **2**

Conjunto de Instruções do IAS

- Transferência de dados
 - Mover dados da memória para registradores

LOAD MQ, M(X) // $MQ \leq Mem[X]$

LOAD M(X) // $AC \leq Mem[X]$

LOAD -M(X) // $AC \leq -(Mem[X])$

LOAD |M(X)| // $AC \leq |Mem[X]|$

Conjunto de Instruções do IAS

- Transferência de dados
 - Mover dados de um registrador para outro
`LOAD MQ // AC <= MQ`
 - Mover dados de registradores para a memória
`STOR M(X) // Mem[X] <= AC`

Conjunto de Instruções do IAS

- Aritméticas

- Realizar operações aritméticas

ADD $M(X)$ // $AC \leq AC + Mem[X]$

ADD $|M(X)|$ // $AC \leq AC + |Mem[X]|$

SUB $M(X)$ // $AC \leq AC - Mem[X]$

SUB $|M(X)|$ // $AC \leq AC - |Mem[X]|$

Conjunto de Instruções do IAS

- Aritméticas

- Realizar operações aritméticas

LSH // AC (0 : 38) <= AC (1 : 39)
 // AC (39) <= 0

RSH // AC (1 : 39) <= AC (0 : 38)
 // AC (0) <= 0

Conjunto de Instruções do IAS

- Aritméticas

- Realizar operações aritméticas

MUL M(X) // AC:MQ \leftarrow MQ x Mem[X]

DIV M(X) // MQ \leftarrow AC / Mem[X]

// AC \leftarrow AC % Mem[X]

Conjunto de Instruções do IAS

- Exemplo: Computar a expressão:

$$(234*3) + (899*23)$$

Conjunto de Instruções do IAS

•Exemplo: $(234*3)+(899*23)$

```
LOAD MQ,M(0x100)    // Carregar o valor 234 em MQ
MUL  M(0x101)        // e multiplicar por 3
LOAD MQ              // Mover o resultado para AC e
STOR M(0x0FF)        // salvar no temporario (0x0FF)
LOAD MQ,M(0x102)    // Carregar o valor 899 em MQ
MUL  M(0x103)        // e multiplicar por 23
LOAD MQ              // Mover o resultado para AC e
ADD  M(0x0FF)        // somar com o temporario.
```

```
00FF 00 00 00 00 00 // Temporario
0100 00 00 00 00 EA // 234
0101 00 00 00 00 03 // 3
0102 00 00 00 03 83 // 899
0103 00 00 00 00 17 // 23
```

Conjunto de Instruções do IAS

- Exemplo: Computar a expressão: $(234 * 3) + (899 * 23)$

LOAD MQ, M(0x100)	
MUL M(0x101)	0000 09 100 0B 101
<hr/>	
LOAD MQ	
STOR M(0x0FF)	0001 0A 000 21 0FF
<hr/>	
LOAD MQ, M(0x102)	
MUL M(0x103)	0002 09 102 0B 103
<hr/>	
LOAD MQ	
ADD M(0x0FF)	0003 0A 000 05 0FF

00FF	00	00	00	00	00	//	Temporario	00FF	00	00	00	00	00
0100	00	00	00	00	EA	//	234	0100	00	00	00	00	EA
0101	00	00	00	00	03	//	3	0101	00	00	00	00	03
0102	00	00	00	03	83	//	899	0102	00	00	00	03	83
0103	00	00	00	00	17	//	23	0103	00	00	00	00	17

Conjunto de Instruções do IAS

- Exemplo: Computar a expressão

```
LOAD MQ,M(0x100)
MUL  M(0x101)
LOAD MQ
STOR M(0x0FF)
LOAD MQ,M(0x102)
MUL  M(0x103)
LOAD MQ
ADD  M(0x0FF)
```

Não é necessário iniciar
a palavra de memória
reservada para o
temporário

00 000 05 0FF

00FF 00 00 00 00 00 // Temporario	00FF 00 00 00 00 00
0100 00 00 00 00 EA // 234	0100 00 00 00 00 EA
0101 00 00 00 00 03 // 3	0101 00 00 00 00 03
0102 00 00 00 03 83 // 899	0102 00 00 00 03 83
0103 00 00 00 00 17 // 23	0103 00 00 00 00 17

Conjunto de Instruções do IAS

- Salto Incondicional
 - Desviar o fluxo de execução

JUMP M(X,0:19) // Salta para a
instrução à esquerda de Mem[X]

JUMP M(X,20:39) // Salta para a
instrução à direita de Mem[X]

Conjunto de Instruções do IAS

- Salto Condicional
 - Desviar o fluxo de execução

JUMP+ M(X,0:19) // Se AC \geq 0,
salta para a instrução à esquerda de
Mem[X], senão executa a próxima
instrução

JUMP+ M(X,20:39) // instr. à dir.

Conjunto de Instruções do IAS

- Exemplo: Computar o fatorial de N:

```
fat = 1;
```

```
i = 1;
```

```
faça
```

```
    fat = fat * i
```

```
    i = i+1;
```

```
enquanto i <= N
```

Conjunto de Instruções do IAS

```
fat = 1;
i = 1;
faça
    fat = fat * i
    i = i+1;
enquanto i <= N
```

Comeco:

```
LOAD MQ,M(0x102) // Carrega fat em MQ
MUL  M(0x103)     // Multiplica MQ por i
LOAD MQ           // Salva resultado em
STOR M(0x102)     // fat.
LOAD M(0x103)     // Carrega i em AC
ADD  M(0x101)     // Incrementa AC e salva
STOR M(0x103)     // resultado em i
LOAD M(0x100)     // Carrega N em AC
SUB  M(0x103)     // AC = AC - i
JUMP+ M(Comeco)   // Salta para Comeco se
Fim              // N-i >= 0
```

```
# N: (N=10)
0100 00 00 00 00 0A
# Constante 1
0101 00 00 00 00 01
# fat
0102 00 00 00 00 01
# i
0103 00 00 00 00 01
```


Conjunto de Instruções do IAS

```
fat = 1;
i = 1;
faça
    fat = fat * i
    i = i+1;
enquanto i <= N
```

Começo:

LOAD MQ,M(0x102)

MUL M(0x103) 0000 09 102 0B 103

LOAD MQ

STOR M(0x102) 0001 0A 000 21 102

LOAD M(0x103)

ADD M(0x101) 0002 01 103 05 101

STOR M(0x103)

LOAD M(0x100) 0003 21 103 01 100

SUB M(0x103)

JUMP+ M(**Começo**) 0004 06 103 **0F 000**

Fim

0005 00 000 00 000

N: (N=10)

0100 00 00 00 00 0A 0100 00 00 00 00 0A

Constante 1

0101 00 00 00 00 01 0101 00 00 00 00 01

fat

0102 00 00 00 00 01 0102 00 00 00 00 01

i

0103 00 00 00 00 01 0103 00 00 00 00 01

Conjunto de Instruções do IAS

Comeco:

```
fat = 1;
```

```
i = 1;
```

```
faça
```

```
fat = fat * i
```

```
LOAD MQ,M(0x102)
```

```
MUL M(0x103) 0000 09 102 0B 103
```

```
LOAD MQ
```

```
STOR M(0x102) 0001 0A 000 21 102
```

```
LOAD M(0x103)
```

```
ADD M(0x101) 0002 01 103 05 101
```

```
STOR M(0x103)
```

```
LOAD M(0x100) 0003 21 103 01 100
```

```
SUB M(0x103)
```

```
JUMP+ M(Comeco) 0004 06 103 0F 000
```

```
Fim
```

Código da operação “**0F**”.

Salta para a instrução à
esquerda da palavra no
endereço **000**

```
0000 00 000 00 000
```

```
0100 00 00 00 00 0A
```

```
00 01 0101 00 00 00 00 01
```

```
00 01 0102 00 00 00 00 01
```

```
00 01 0103 00 00 00 00 01
```

Conjunto de Instruções do IAS

- Exemplo: Somar os valores de um vetor com N números. Suponha que o vetor comece no endereço 0x070 e que $N > 0$.

```
soma=0  
i=0  
faça  
    soma = soma + vetor[i]  
    i = i+1;  
enquanto i < N
```

Conjunto de Instruções do IAS

- Modificação de endereços
 - Modificar o campo endereço de uma instrução na memória

STOR M(X, 8:19)

// Mem[X](8:19) <= AC(28:39)

STOR M(X, 28:39)

// Mem[X](28:39) <= AC(28:39)

Conjunto de Instruções do IAS

Começo:

```
LOAD M(0x069) // Carrega a base do vetor,  
ADD M(0x067) // soma com i e armazena o  
STOR M(soma) // endereço da instrução "soma"
```

soma=0

i=0

faça

soma=soma+vetor[i]

i = i+1;

enquanto i < N

```
// Realiza a soma -- Soma = soma + vetor[i]  
LOAD M(0x068) // Carrega a variavel soma,
```

soma:

```
ADD M(0x000) // soma com Vetor[i] e  
STOR M(0x068) // armazena em soma.
```

```
// Atualiza i -- i = i+1
```

```
LOAD M(0x067) // Carrega i em AC,  
ADD M(0x065) // soma com a constante 1, e  
STOR M(0x067) // armazena em i
```

```
// Enquanto i < N
```

```
LOAD M(0x066) // Carrega (N-1)  
SUB M(0x067) // AC = (N-1) - i  
JUMP+ M(Comeco) // Salta para comeco se  
// (N-1) - i >= 0
```

Fim

```
0065 00 00 00 00 01 // Constante 1  
0066 00 00 00 00 27 // N-1 (N=40)  
0067 00 00 00 00 00 // i  
0068 00 00 00 00 00 // Soma  
0069 00 00 00 00 70 // Base do vetor  
0070 00 00 00 00 01 // 1ª posicao do vetor
```

Conjunto de Instruções do IAS

Comeco:

LOAD	M(0x069)	
ADD	M(0x067)	0000 01 069 05 067
STOR	M(soma)	

soma=0

i=0

faça

soma=soma+vetor[i]

i = i+1;

enquanto i < N

LOAD	M(0x068)	0001 12 002 01 068
------	----------	--------------------

soma:

ADD	M(0x000)	
STOR	M(0x068)	0002 05 000 21 068

LOAD	M(0x067)	
ADD	M(0x065)	0003 01 067 05 065
STOR	M(0x067)	

LOAD	M(0x066)	0004 21 067 01 066
SUB	M(0x067)	
JUMP+	M(Comeco)	0005 06 067 0F 000

Fim		0006 00 00 00 00 00
0065	00 00 00 00 01	0065 00 00 00 00 01
0066	00 00 00 00 27	0066 00 00 00 00 28
0067	00 00 00 00 00	0067 00 00 00 00 00
0068	00 00 00 00 00	0068 00 00 00 00 00
0069	00 00 00 00 70	0069 00 00 00 00 70
0070	00 00 00 00 01	0070 00 00 00 00 01

Conjunto de Instruções do IAS

Começo:

LOAD	M(0x069)						
ADD	M(0x067)	0000	01	069	05	067	
STOR	M(soma)						

soma=0

i=0

faça

Instrução de soma à esquerda da palavra no endereço 002

LOAD	M(0x068)	0001	12	002	01	068	
soma:							
ADD	M(0x000)						
STOR	M(0x068)	0002	01	000	21	068	

LOAD	M(0x067)						
ADD	M(0x065)	0003	01			065	
STOR	M(0x067)						

Código da operação "12". Move os 12 bits à direita de AC para o campo endereço da instrução à esquerda da palavra no endereço 002

Fin
0065
0066
0067
0068
0069
0070

Conjunto de Instruções do IAS

- Exercício: buscar o primeiro número negativo em um vetor.

```
i = 0;  
enquanto(vetor[i] >= 0)  
{  
    i = i+1;  
}
```