

INSTITUTO FEDERAL
Sul-rio-grandense

Câmpus
Venâncio Aires

Arquitetura e Organização de Computadores

Professor: Fernando Luís Herrmann

E-mail: fernandoherrmann@ifsul.edu.br



Material de aula:

<https://github.com/herrmannfl/tads-aoc-2022>



Simulador Neander



Programa

P	End.	Dado	Mnemônico
→	0	0	NOP
	1	0	NOP
	2	0	NOP
	3	0	NOP
	4	0	NOP
	5	0	NOP
	6	0	NOP
	7	0	NOP
	8	0	NOP
	9	0	NOP
	10	0	NOP
	11	0	NOP
	12	0	NOP
	13	0	NOP
	14	0	NOP
	15	0	NOP
	16	0	NOP

BP: 255 [0]: 0

Neander

Arquivo Editar Visualizar Executar ?

AC:

PC:

Execução:

Acessos:

Instruções:

Instrução:

Reg.Instrução: 0

Mnemônico: NOP

Ok.

Mnemônicos

NOP	00	ADD	48 end	JMP	128 end
STA	16 end	OR	64 end	JN	144 end
LDA	32 end	AND	80 end	JZ	160 end
		NOT	96	HLT	240

Neander

Versão 2.1.2
Julho 2002
●●●●●●●●

Autores: Raul Fernando Weber
Taisy Silva Weber

Versão: Fabio Augusto Dal Castel
Win32

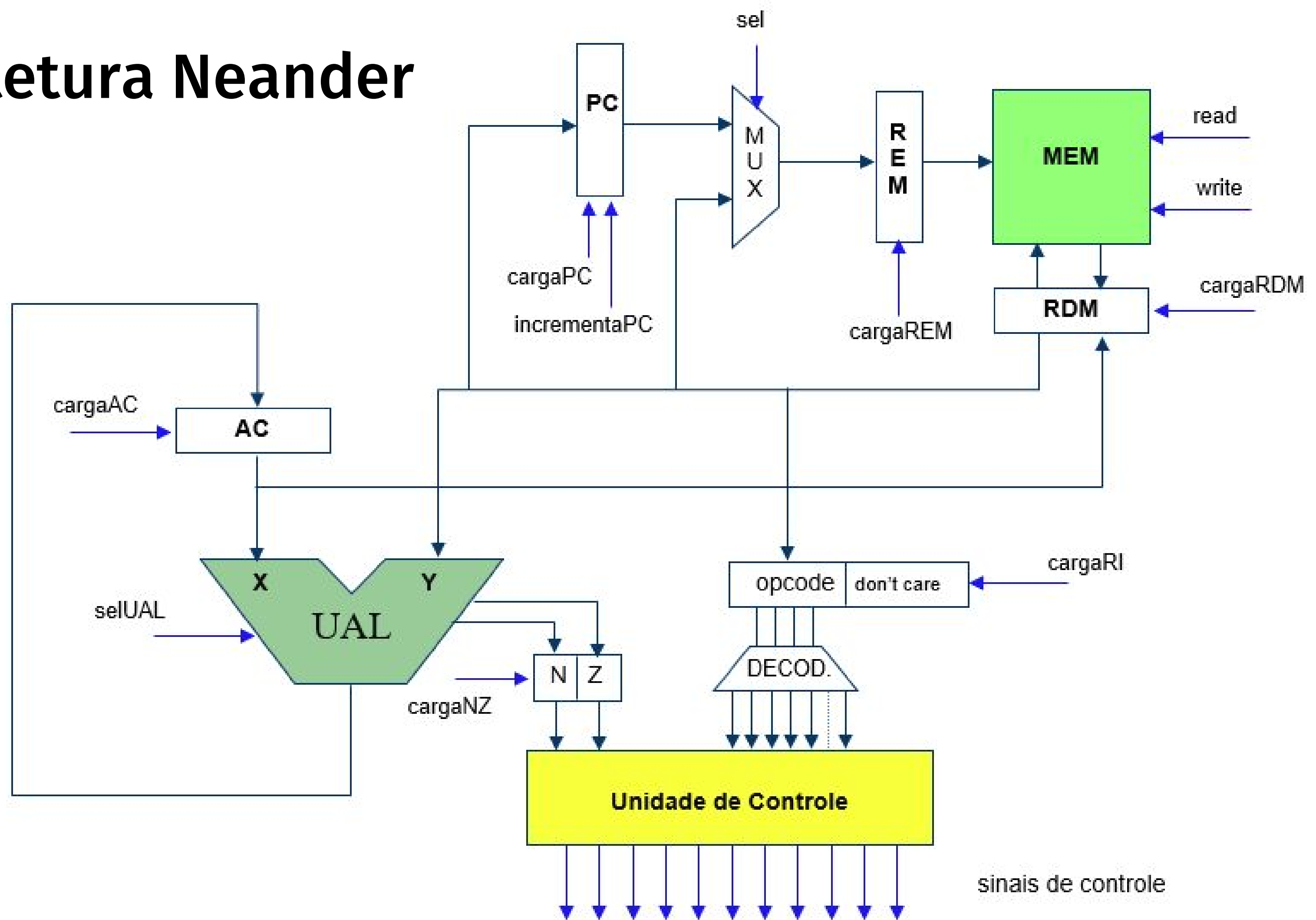
Ok

Dados

End.	Dado
128	0
129	0
130	0
131	0
132	0
133	0
134	0
135	0
136	0
137	0
138	0
139	0
140	0
141	0
142	0
143	0
144	0

[128]: 0

Arquitetura Neander



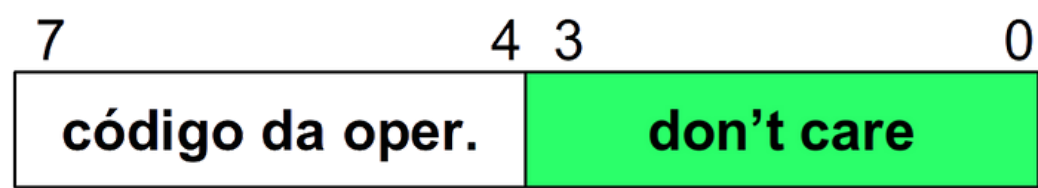
Características gerais

- Largura de dados e endereços de 8 bits
- Dados representados em complemento de 2
- 1 acumulador de 8 bits (AC)
- 1 apontador de programa de 8 bits (PC)
- 1 registrador de estado com 2 códigos de condição: negativo (N) zero (Z)
- Memória de 256 posições (endereços) x 8 bits

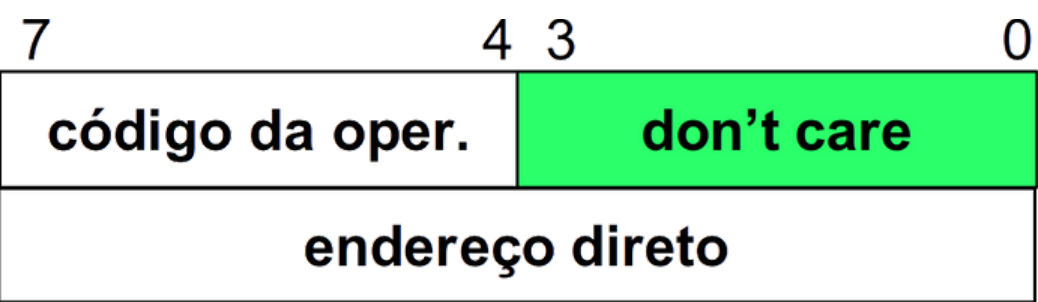
Conjunto de Instruções NEANDER

código	instrução	comentário
0000	NOP	Nenhuma operação
0001	STA end	MEM(end) ← AC
0010	LDA end	AC ← MEM(end)
0011	ADD end	AC ← MEM(end) + AC
0100	OR end	AC ← MEM(end) OR AC
0101	AND end	AC ← MEM(end) AND AC
0110	NOT	AC ← NOT AC
1000	JMP end	PC ← end
1001	JN end	IF N=1 THEN PC ← end
1010	JZ end	IF Z=1 THEN PC ← end
1111	HLT	para processamento

Instruções com um byte:
NOP, NOT



Instruções com dois bytes:
STA, LDA, ADD, OR, AND,
JMP, JN, JZ



Mnemônicos					
NOP	00	ADD	48 end	JMP	128 end
STA	16 end	OR	64 end	JN	144 end
LDA	32 end	AND	80 end	JZ	160 end
		NOT	96	HLT	240