

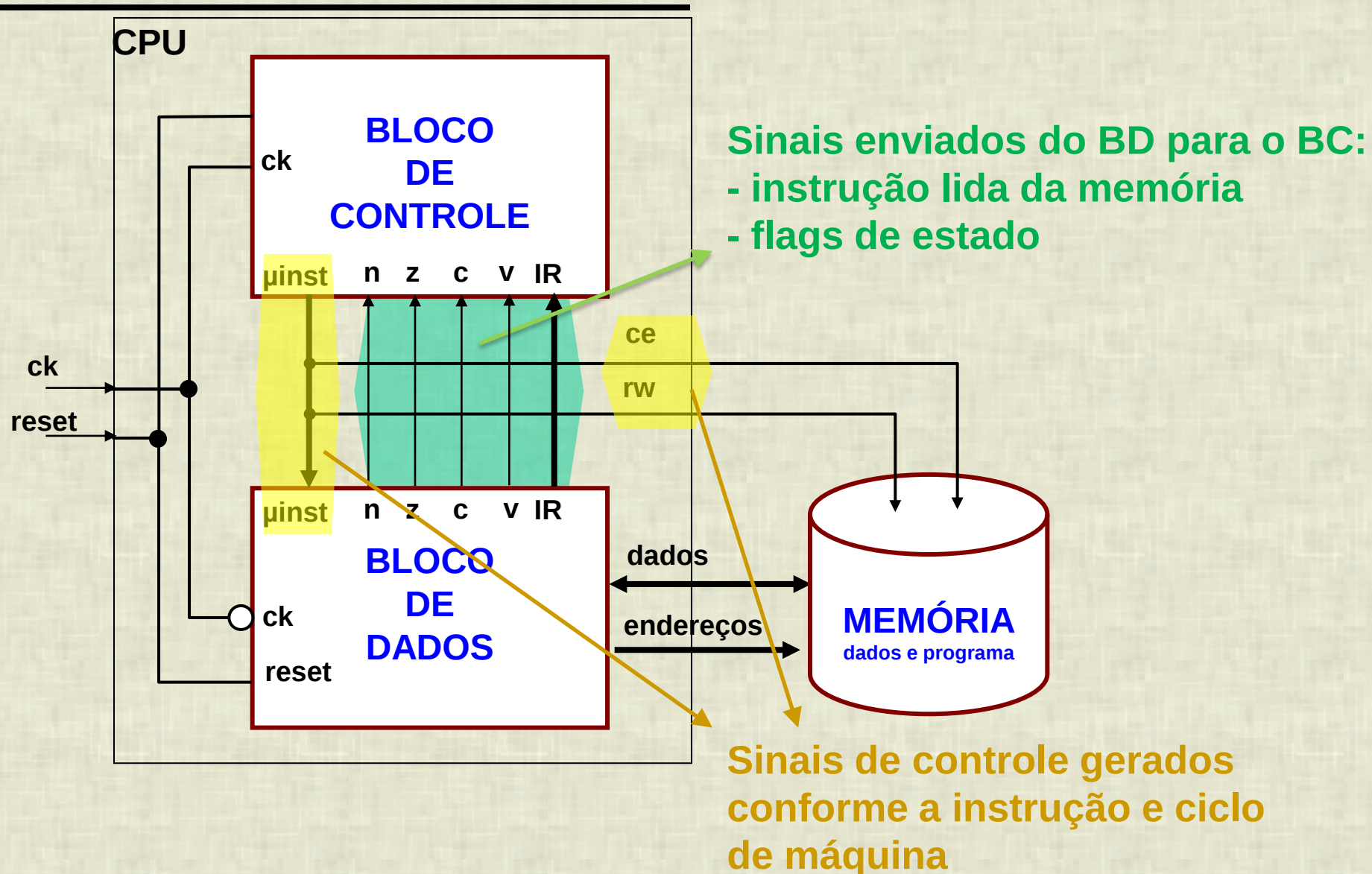
ORGANIZAÇÃO E ARQUITETURA DE COMPUTADORES I

Arquitetura Cleópatra

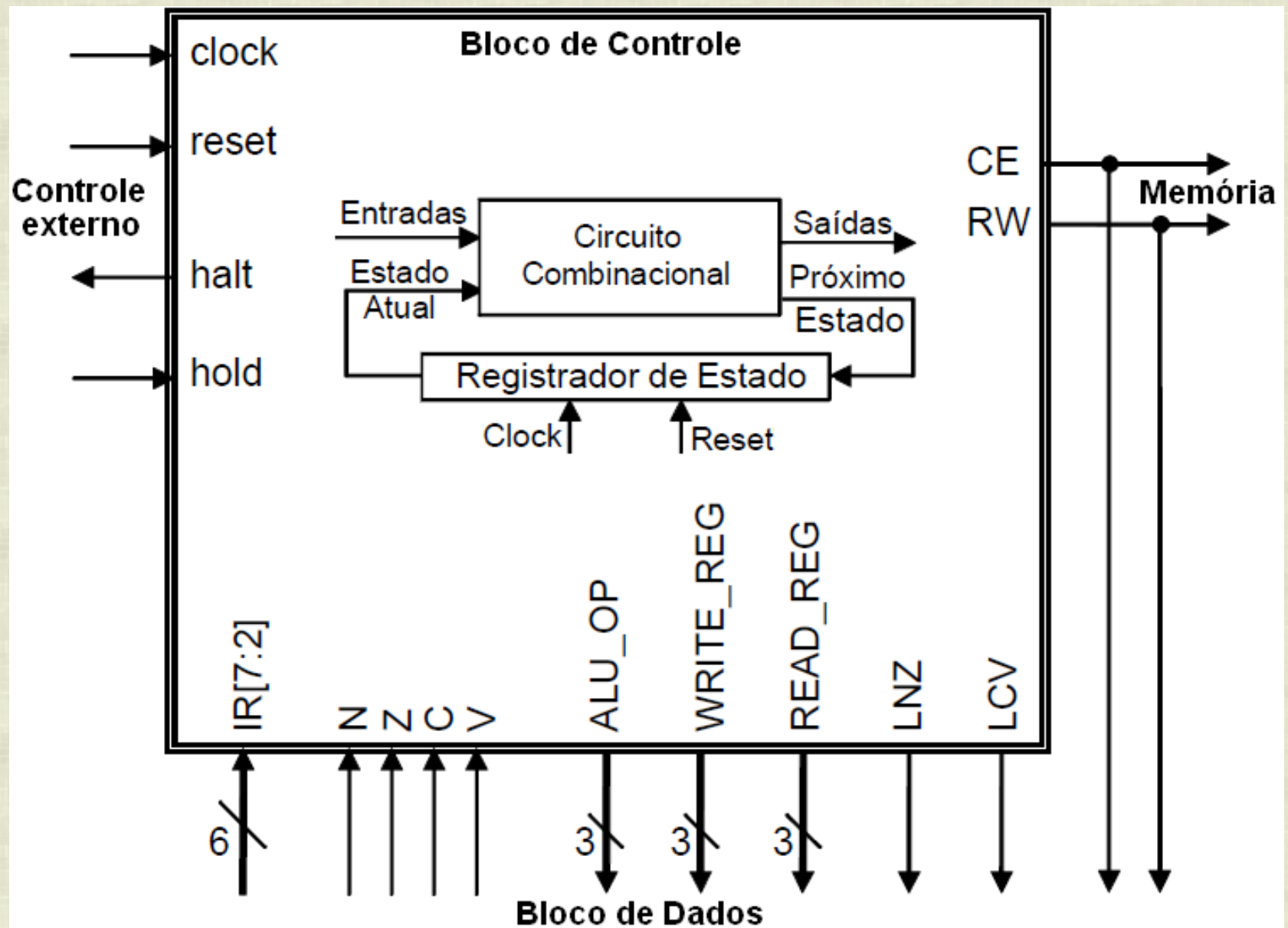
Bloco de Controle (Instruções e Microinstruções)

prof. Dr. César Augusto M. Marcon
prof. Dr. Edson Ifarraguirre Moreno

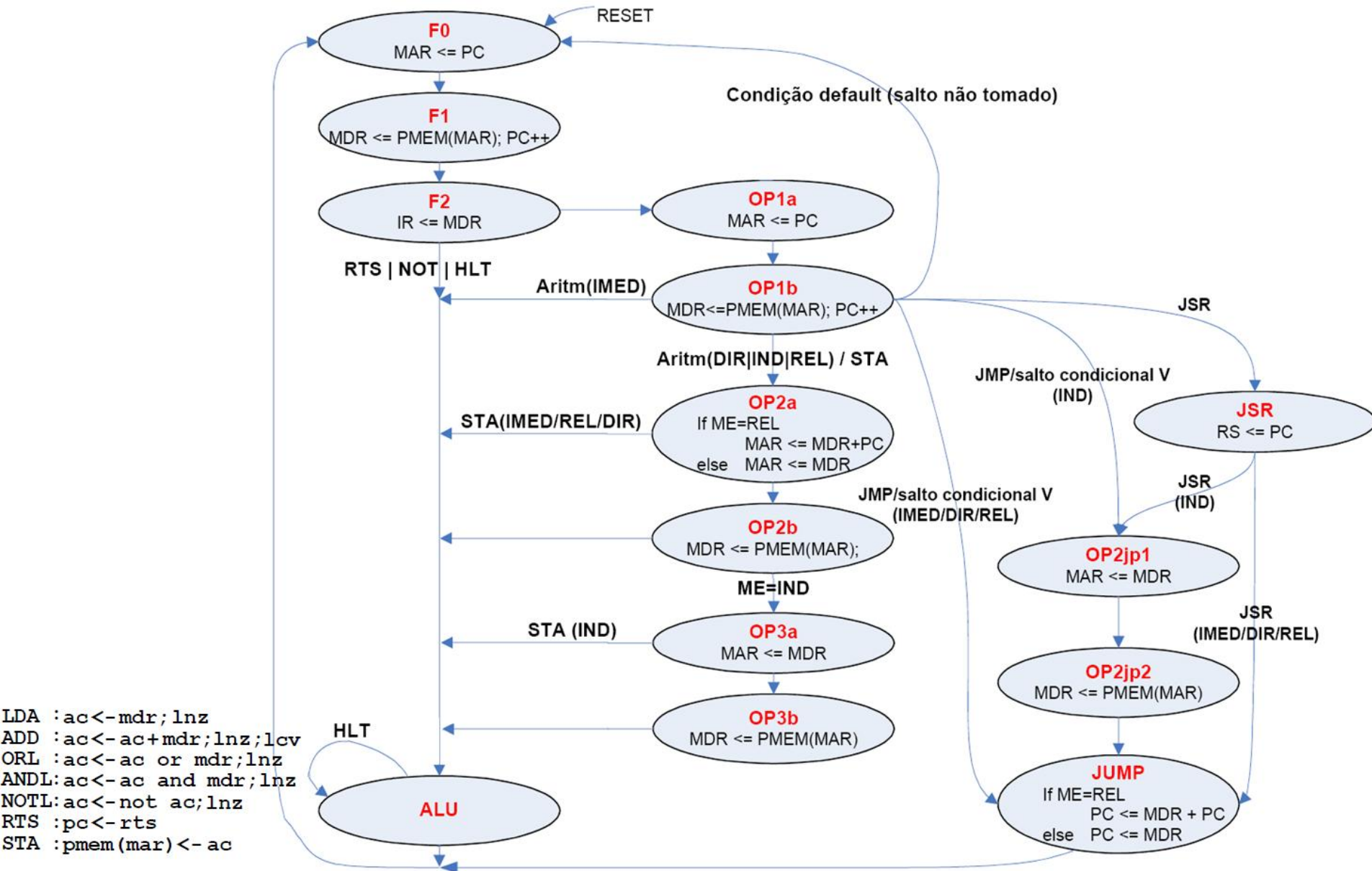
Comunicações entre Bloco de Dados e Bloco de Controle



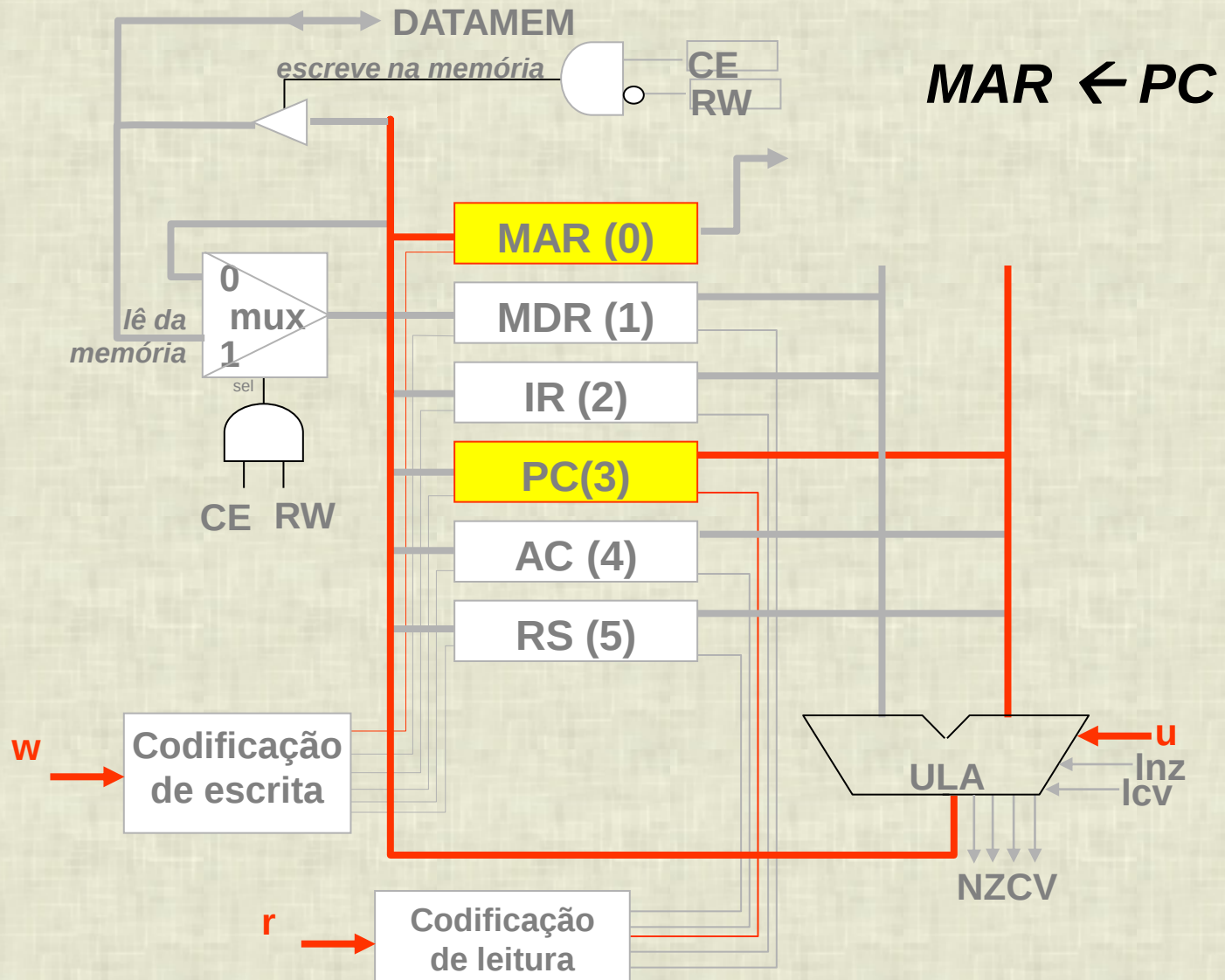
Interface de Entrada e Saída do Bloco de Controle



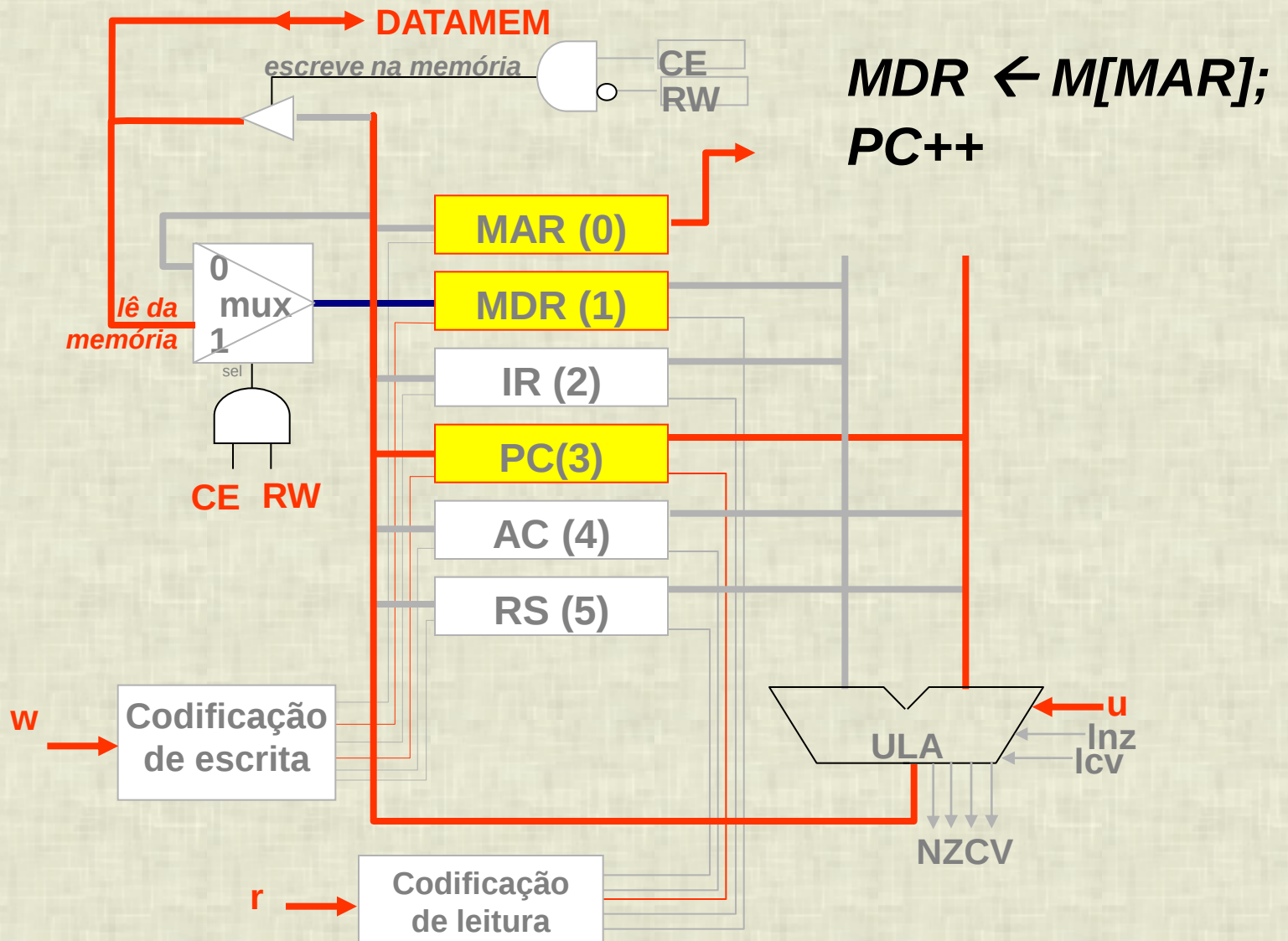
Máquina de Estados Finita



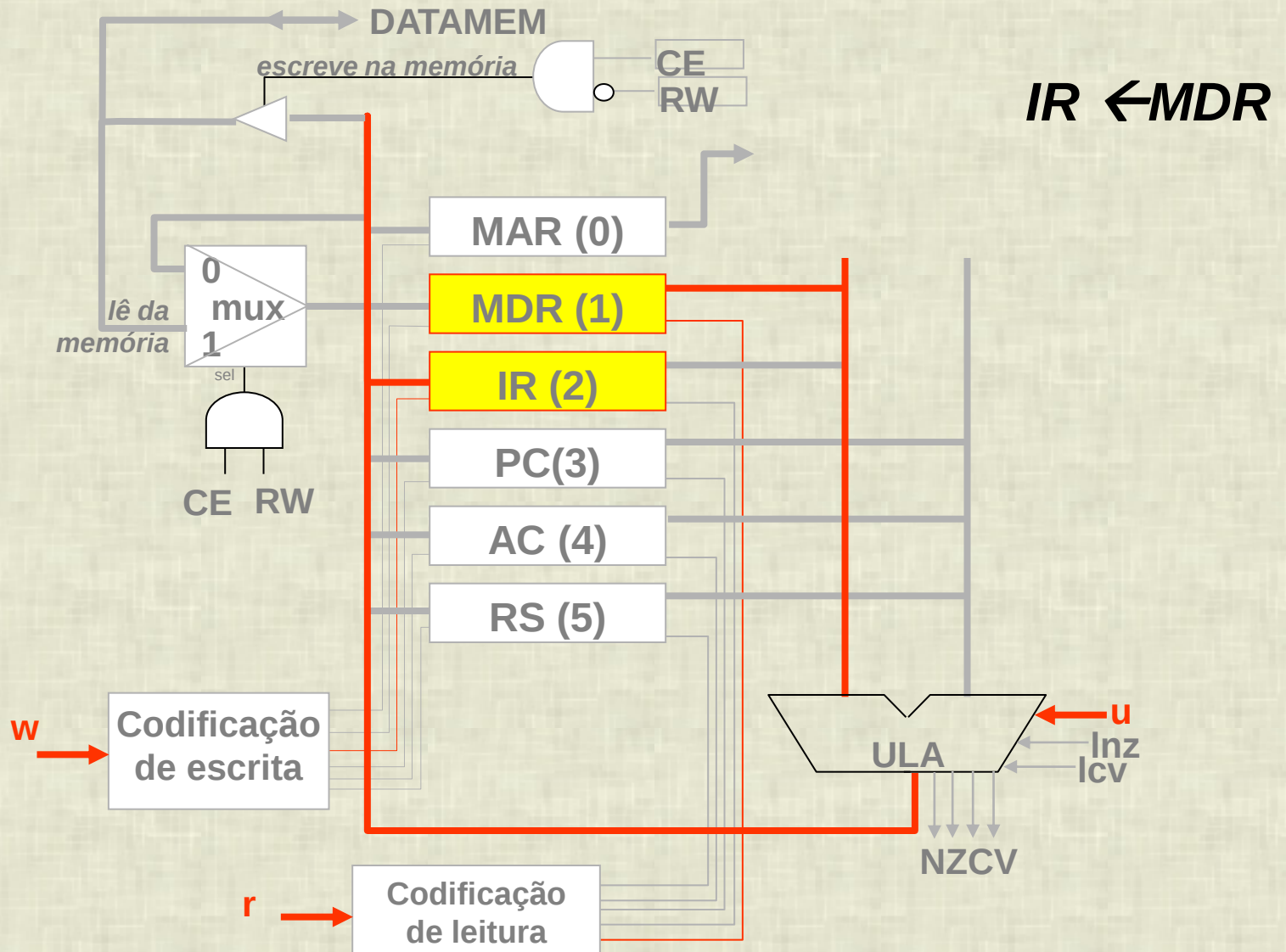
FETCH 1/3



FETCH 2/3



FETCH 3/3



Exercícios

- 1. Complete a tabela com as microinstruções correspondentes a cada instrução, sendo 3Ch uma constante em hexadecimal e VAR uma variável de memória**

[illegible]

Resposta de Exercícios

- Complete a tabela com as microinstruções correspondentes a cada instrução, sendo 3Ch uma constante em hexadecimal e VAR uma variável de memória

Microinstruções	Instruções
MAR $\leftarrow$ PC	LDA #3CH
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
AC $\leftarrow$ MDR , LNZ	
MAR $\leftarrow$ PC	ADD VAR
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
MAR $\leftarrow$ MDR	
MDR $\leftarrow$ M[MAR]	
AC $\leftarrow$ AC + MDR , LNZ , LCV	

Exercícios

2. Dado o conjunto de microinstruções abaixo, associe a instrução correspondente à direita. Caso a instrução tenha operando e este seja uma constante, coloque o valor 33h, caso o operando seja uma variável coloque o label vv

Microinstruções	Instruções
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
AC $\leftarrow$ NOT AC, LNZ	
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
MAR $\leftarrow$ MDR	
MDR $\leftarrow$ M[MAR]	
MAR $\leftarrow$ MDR	
MDR $\leftarrow$ M[MAR]	
AC $\leftarrow$ AC OR MDR, LNZ, LCV	

Resposta de Exercícios

2. Dado o conjunto de microinstruções abaixo, associe a instrução correspondente à direita. Caso a instrução tenha operando e este seja uma constante, coloque o valor 33h, caso o operando seja uma variável coloque o label vv

Microinstruções	Instruções
MAR $\leftarrow$ PC	NOT
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
AC $\leftarrow$ NOT AC , LNZ	
MAR $\leftarrow$ PC	OR vv , I
MDR $\leftarrow$ M[MAR] , PC++	
IR $\leftarrow$ MDR	
MAR $\leftarrow$ PC	
MDR $\leftarrow$ M[MAR] , PC++	
MAR $\leftarrow$ MDR	
MDR $\leftarrow$ M[MAR]	
MAR $\leftarrow$ MDR	
MDR $\leftarrow$ M[MAR]	
AC $\leftarrow$ AC OR MDR , LNZ , LCV	

Exercícios

3. Dado o programa ao lado (já com o código objeto) e a tabela de instruções/microinstruções, preencha os registradores e flags e a área de dados da Cleópatra

Área de programa

```
00: 40 5B      LDA    #5BH
02: 58 75      ADD    A, I
04: EC 01      JV     1, R
06: 00         NOT
07: F0         HLT
```

Área de dados

```
.org #75H
A: DB #B      75:
B: DB 78H     76:
```

CK	MAR	MDR	IR	PC	AC	RS	N	Z	C	V	Microinstruções	Instruções
1											MAR $\leftarrow$ PC	LDA #5BH
2											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
3											IR $\leftarrow$ MDR	
4											MAR $\leftarrow$ PC	
5											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
6											AC $\leftarrow$ MDR, LNZ	
7											MAR $\leftarrow$ PC	ADD 75H,I
8											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
9											IR $\leftarrow$ MDR	
10											MAR $\leftarrow$ PC	
11											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
12											MAR $\leftarrow$ MDR	
13											MDR $\leftarrow$ M[MAR]	
14											MAR $\leftarrow$ MDR	
15											MDR $\leftarrow$ M[MAR]	JV 1,R
16											AC $\leftarrow$ AC + MDR, LNZ, LCV	
17											MAR $\leftarrow$ PC	
18											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
19											IR $\leftarrow$ MDR	
20											MAR $\leftarrow$ PC	
21											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
22											PC $\leftarrow$ PC + MDR	
23											MAR $\leftarrow$ PC	HLT
24											MDR $\leftarrow$ M[MAR], PC $\leftarrow$ PC+1	
25											IR $\leftarrow$ MDR	

Resposta de Exercícios

3. Dado o programa ao lado (já com o código objeto) e a tabela de instruções/microinstruções, preencha os registradores e flags e a área de dados da Cleópatra

Área de programa

Área de dados

```
00: 40 5B      LDA    #5BH
02: 58 75      ADD    A,I
04: EC 01      JV     1,R
06: 00         NOT
07: F0         HLT
```

.org #75H

A: DB #B 75: **76**

```
B: DB 78H      76: 78
```

[illegible]

Exercícios

4. Qual seria o tempo de execução do programa apresentado no exercício anterior (em segundo), se a Cleópatra tivesse uma frequência de operação de 10MHz?

Resposta de Exercícios

4. Qual seria o tempo de execução do programa apresentado no exercício anterior (em segundo), se a Cleópatra tivesse uma frequência de operação de 10MHz?

$$f = 10 \text{ MHz} \rightarrow T = 0,1\mu\text{s} \rightarrow \text{tempo de execução} = 25 * 0,1\mu\text{s} = 2,5\mu\text{s}$$

Exercícios

5. Dado o programa em linguagem C apresentado abaixo, faça:

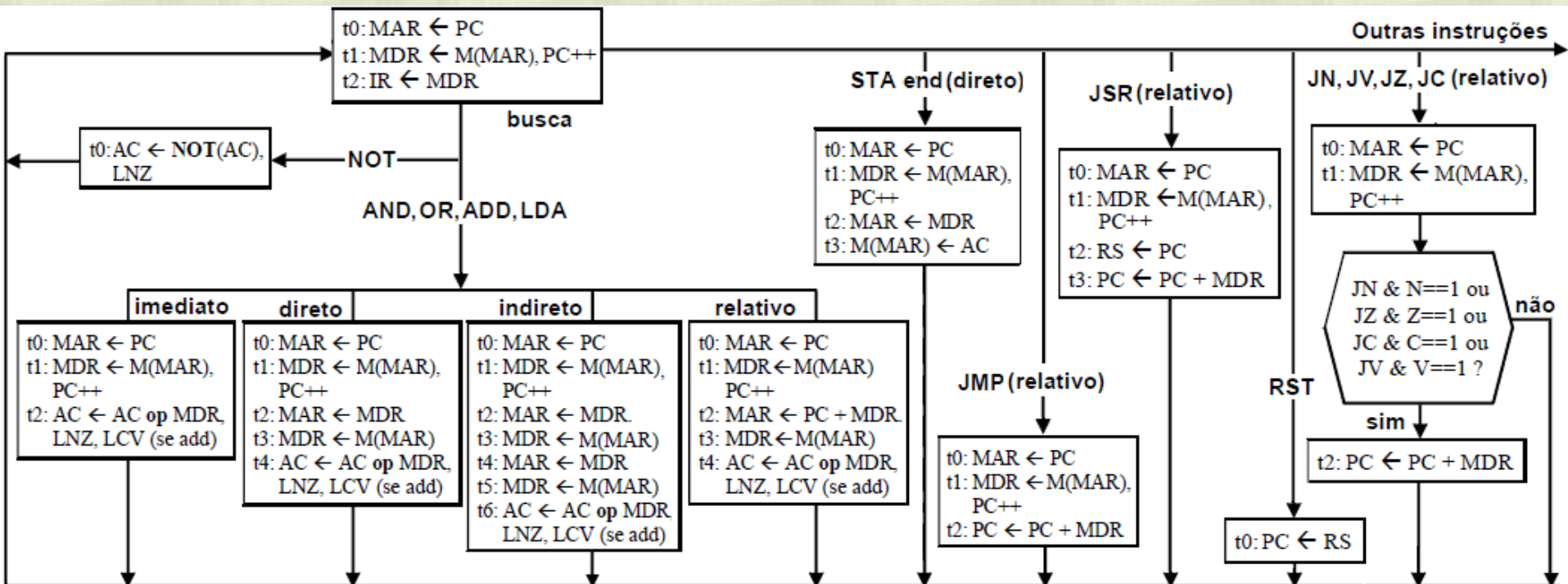
- A compilação deste para o assembly da arquitetura Cleópatra
- A codificação do assembly para a linguagem de máquina em hexadecimal, tanto para a área de dados, quanto para a área de código
- Uma fórmula que caracterize o tempo de execução do programa em função das variáveis de controle de laços
- A estimativa de qual deve ser a frequência de operação do relógio para que este programa consiga ser executado em um tempo praticamente igual a 1us
- A tradução das cinco primeiras instruções assembly para o microprograma (conjunto de microinstruções) correspondente
- A associação das cinco microinstruções com suas respectivas microoperações
- O preenchimento de todos os registradores e flip-flops devido a primeira passagem de cada uma das cinco microinstruções

```
#define TAM 10
int vetA[TAM];

void main()
{
    for(int i=0; i < TAM - 1; i++)
    {
        for(int k=i; k < TAM-1; k++)
        {
            if(vetA[i] > vetA[k])
            {
                int aux = vetA[i];
                vetA[i] = vetA[k];
                vetA[k] = aux;
            }
        }
    }
}
```

EXTRAS

Máquina de Estados Finita (parcial e em blocos)



3.3.1 Programa exemplo

Endereço de memória	Instruções	Código objeto	
		Em Binário	Em Hexadecimal
0 e 1	LDA #33H	01000000 00110011	40 33
2 e 3	ADD 83,D	01010100 10000011	54 83
4 e 5	STA 18H,D	00100100 00011000	24 18
Endereço	Dados	Dados em Binário	Dados em Hexa
18	00H	00000000	00
...	...	...	...
83	0D6H	11010110	D6

CK	DIN	DOUT	ALU_op	write_reg	read_reg	lnz	lcv	ce	rw	ADDR ESS	MDR	IR	PC	AC	RS	N Z C V	Microcódigo	Ciclo de Máquina
0	--	--	7	0	3	0	0	0	0	00	00	00	00	00	00	0 0 0 0	MAR ← PC.	Busca LDA ,#
1	40	--	1	6	3	0	0	1	1	00	40	00	01	00	00	0 0 0 0	MDR ← PMEM(MAR), PC++.	
2	--	--	4	2	1	0	0	0	0	00	40	40	01	00	00	0 0 0 0	IR ← MDR.	
3	--	--	7	0	3	0	0	0	0	01	40	40	01	00	00	0 0 0 0	MAR ← PC.	Execução LDA ,#
4	33	--	1	6	3	0	0	1	1	01	33	40	02	00	00	0 0 0 0	MDR ← PMEM(MAR), PC++.	
5	--	--	4	4	1	1	0	0	0	01	33	40	02	33	00	0 0 0 0	AC ← MDR, LNZ.	
6	--	--	7	0	3	0	0	0	0	02	33	40	02	33	00	0 0 0 0	MAR ← PC.	Busca ADD ,D
7	54	--	1	6	3	0	0	1	1	02	54	40	03	33	00	0 0 0 0	MDR ← PMEM(MAR), PC++.	
8	--	--	4	2	1	0	0	0	0	02	54	54	03	33	00	0 0 0 0	IR ← MDR.	
9	--	--	7	0	3	0	0	0	0	03	54	54	03	33	00	0 0 0 0	MAR ← PC.	Execução ADD ,D
10	83	--	1	6	3	0	0	1	1	03	83	54	04	33	00	0 0 0 0	MDR ← PMEM(MAR), PC++.	
11	--	--	4	0	1	0	0	0	0	83	83	54	04	33	00	0 0 0 0	MAR ← MDR.	
12	D6	--	7	1	0	0	0	1	1	83	D6	54	04	33	00	0 0 0 0	MDR ← PMEM(MAR).	
13	--	--	0	4	6	1	1	0	0	83	D6	54	04	09	00	0 0 1 0	AC ← AC + MDR, LNZ, LCV.	Busca STA ,D
14	--	--	7	0	3	0	0	0	0	04	D6	54	04	09	00	0 0 1 0	MAR ← PC.	
15	24	--	1	6	3	0	0	1	1	04	24	54	05	09	00	0 0 1 0	MDR ← PMEM(MAR), PC++.	
16	--	--	4	2	1	0	0	0	0	04	24	24	05	09	00	0 0 1 0	IR ← MDR.	
17	--	--	7	0	3	0	0	0	0	05	24	24	05	09	00	0 0 1 0	MAR ← PC.	Execução STA ,D
18	18	--	1	6	3	0	0	1	1	05	18	24	06	09	00	0 0 1 0	MDR ← PMEM(MAR), PC++.	
19	--	--	4	0	1	0	0	0	0	18	18	24	06	09	00	0 0 1 0	MAR ← MDR.	
20	09	09	7	7	4	0	0	1	0	18	18	24	06	09	00	0 0 1 0	PMEM(MAR) ← AC.	