



Nome: \_\_\_\_\_

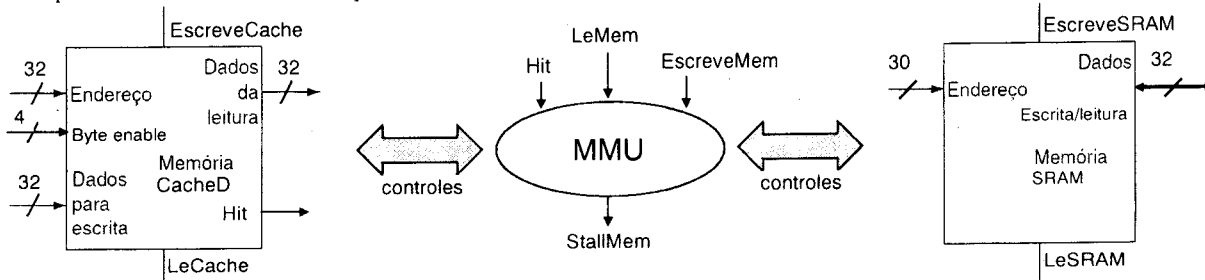
GABARITO

Matrícula: \_\_\_\_\_

□□/□□□□□□□□

## Prova 2

1)(7.0) Hoje em dia, um dos maiores gargalos dos sistemas computacionais é o acesso à memória RAM. O uso da memória cache minimiza esse problema pelo uso de um sistema de hierarquia. Considere uma memória principal SRAM de 4 GiB, que necessita sempre 4 ciclos de clock (independente da frequência) para cada acesso (leitura ou escrita) de uma word (32 bits). Uma memória cache de dados de tamanho convencional de 1 kiB, com bloco de tamanho 4 words, que necessita apenas 1 ciclo de clock para cada acesso. Considere que nunca ocorram falhas na memória de instruções.



Obs: os 4 bits do 'Byte enable' selecionam a combinação dos bytes dos 'Dados para escrita' a serem escritos na memória.

a) (3.0) Projete uma MMU (*Memory Management Unit*) simples (máquina de estados) que receba um sinal de hit da cache de dados diretamente mapeada, gere um sinal de StallMem para o bloco de controle do processador, e faça o tratamento das falhas, tanto para leitura quanto para escrita (*write-through*) na memória SRAM.

b) (2.0) Modifique adequadamente o Caminhos de Dados e o Bloco Controlador do MIPS MULTICICLO fornecidos em anexo de forma a implementar adequadamente as instruções: lw, lb, sw e sb

c) (2.0) Modifique adequadamente o Caminhos de Dados e o Bloco Controlador do MIPS PIPELINE fornecidos em anexo de forma a implementar adequadamente as instruções: lw, lb, sw e sb

2)(4.0) Considerando o uso da estrutura de memória da questão 1) e os seguintes tempos de atraso das unidades operativas do caminho de dados:

| Unidade                                    | Tempo |
|--|-------|
| Operações com a ULA                        | 150ps |
| Leitura/Escrita no Banco de Registradores  | 50ps  |
| Acesso à memória cache de instruções/dados | 200ps |

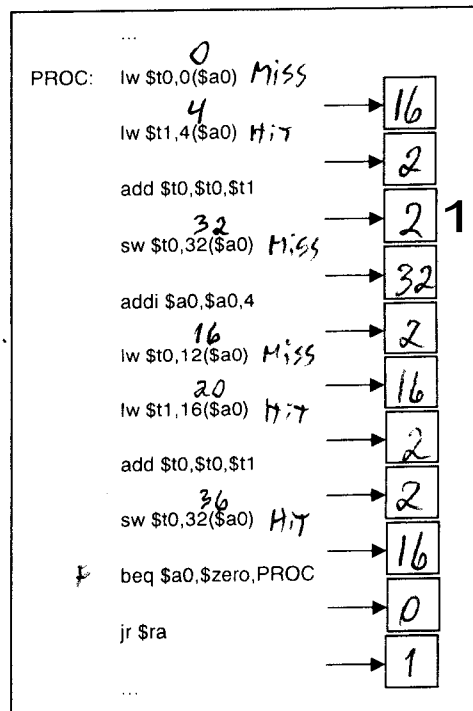
Considere \$a0=0, e que a memória cache de dados esteja vazia no início da execução do procedimento PROC.

a) (1.0) Para a implementação uniciclo, qual será a maior frequência de clock utilizável?

b) (1.0) Para a implementação multiciclo vista em aula, qual será a maior frequência de clock utilizável? Qual o tempo necessário para a execução deste procedimento? jr necessita 3 ciclos e addi necessita 4 ciclos.

c) (1.0) Qual o tempo de execução da implementação pipeline, se todos os hazards e falhas da cache forem tratados apenas com inserção de bolhas? Indique no espaço reservado ao lado, o número de bolhas necessário (obs.: se não houver necessidade de bolha coloque 'zero'). Considere que os registradores podem ser lidos e escritos no mesmo ciclo, que o branch previsto como não tomado é avaliado na etapa EX, e a instrução jr necessita 2 ciclos.

d) (1.0) Qual o tempo de execução para implementação em pipeline, se os hazards forem tratados eficientemente pelo processador com forwarding e/ou inserção de bolhas? Preencha o pipeline esquemático na folha em anexo indicando as instruções, bolhas e forwards necessários. Considere que os registradores podem ser escritos e lidos no mesmo ciclo, que o branch previsto como não tomado é avaliado na etapa ID, e a instrução jr necessita 2 ciclos.



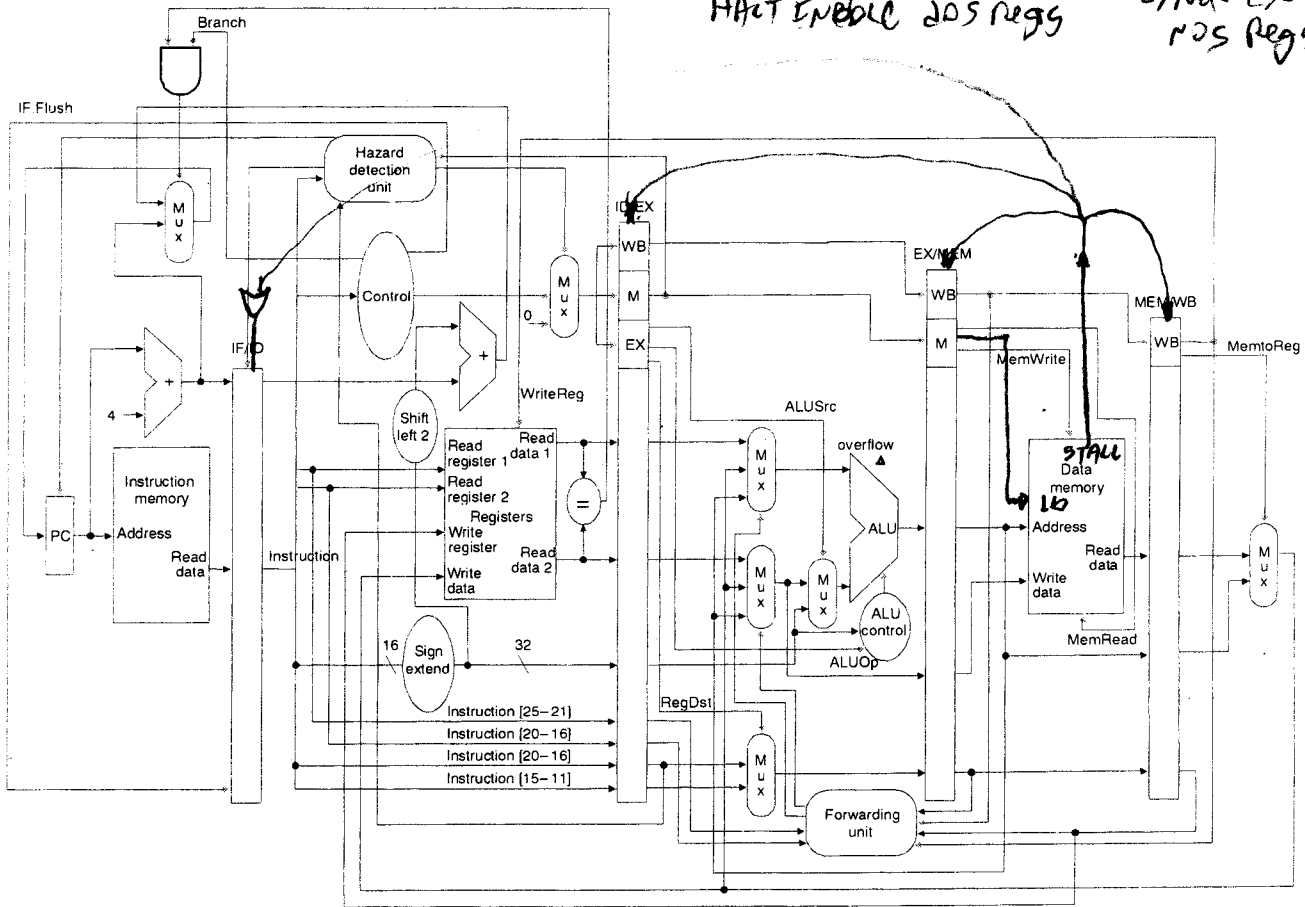
Boa Sorte!!!

No pipeline desconsidere o tempo de latência



STALL=1  
 ↳ NÃO ESCREVER  
 NOS REGS

HART ENABLE DOS REGS



ESTRUTURA DA MEMÓRIA DE DADOS  
 CONFORME DESCRITO  
 NA FOLHA.

| Instrução | Linhas de controle do estágio de cálculo de endereço/execução |        |        |         |  |
|-----------|---|--------|--------|---------|--|
|           | RegDst  | OpALU1 | OpALU0 | OrigALU |  |
| Formato R | 1   | 1      | 0      | 0       |  |
| lw        | 0   | 0      | 0      | 1       |  |
| sw        | X   | 0      | 0      | 1       |  |
| beq       | X   | 0      | 1      | 0       |  |
| lb        | 0   | 0      | 0      | 1       |  |
| sb        | X   | 0      | 0      | 1       |  |

051

| Instrução | Linhas de controle do estágio de acesso à memória |       |             |    |
|-----------|---|-------|-------------|----|
|           | Branch  | LeMem | Escreve Mem | lb |
| Formato R | 0   | 0     | 0           | 0  |
| lw        | 0   | 1     | 0           | 0  |
| sw        | 0   | 0     | 1           | 0  |
| beq       | 1   | 0     | 0           | 0  |
| lb        | 0   | 1     | 0           | 1  |
| sb        | 0   | 0     | 1           | 1  |

| Instrução | Linhas de controle do estágio de escrita do resultado |              |
|-----------|---|--------------|
|           | Escreve Reg   | Mem para Reg |
| Formato R | 1   | 0            |
| lw        | 1   | 1            |
| sw        | 0   | X            |
| beq       | 0   | X            |
| lb        | 1   | 1            |
| sb        | 0   | X            |



Universidade de Brasília

Departamento de Ciência da Computação

Disciplina: CIC 116394 – Organização e Arquitetura de Computadores

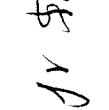
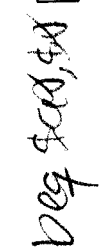
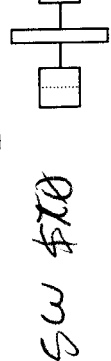
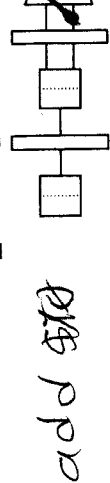
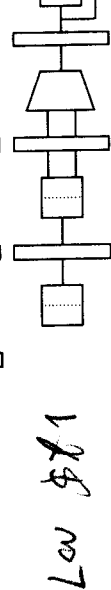
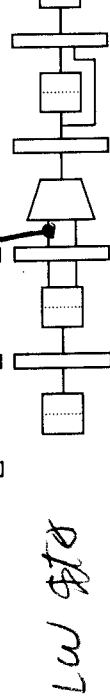
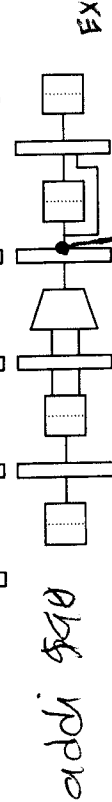
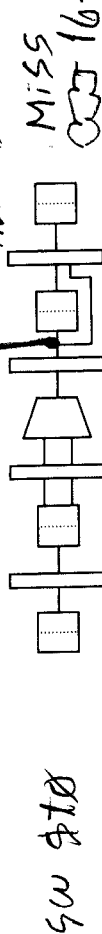
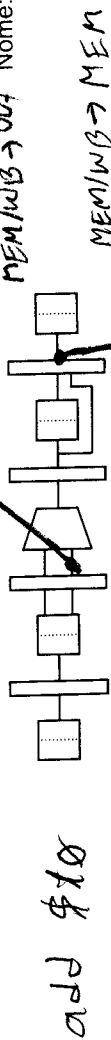
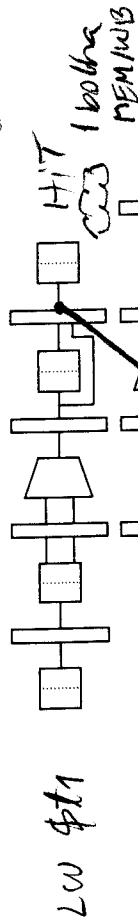
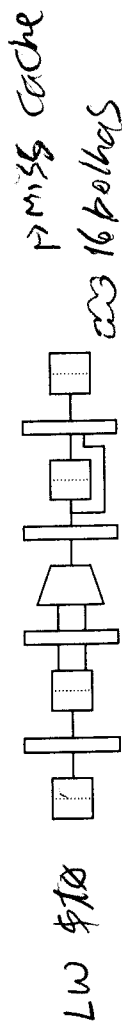
Prof. Marcus Vinicius Lamar

201411

Matrícula:

GABRIÃO

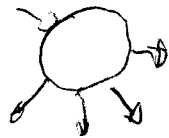
Nome:





# SINAIS DE SAÍDA DOS ESTADOS

INÍCIO:



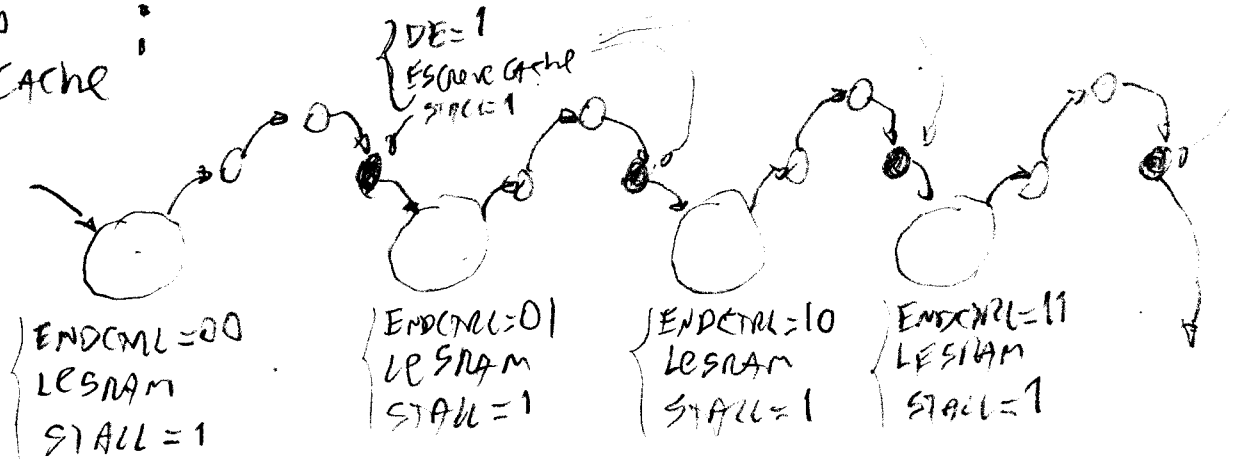
ENDCTRL=00  
DE=X  
ESCREVE CACHE=0  
LE CACHE=0  
STALL=0

ESCREVE SRAM=0  
LE SRAM=0

ESCREVE WORD NA CACHE → 1 CICLO

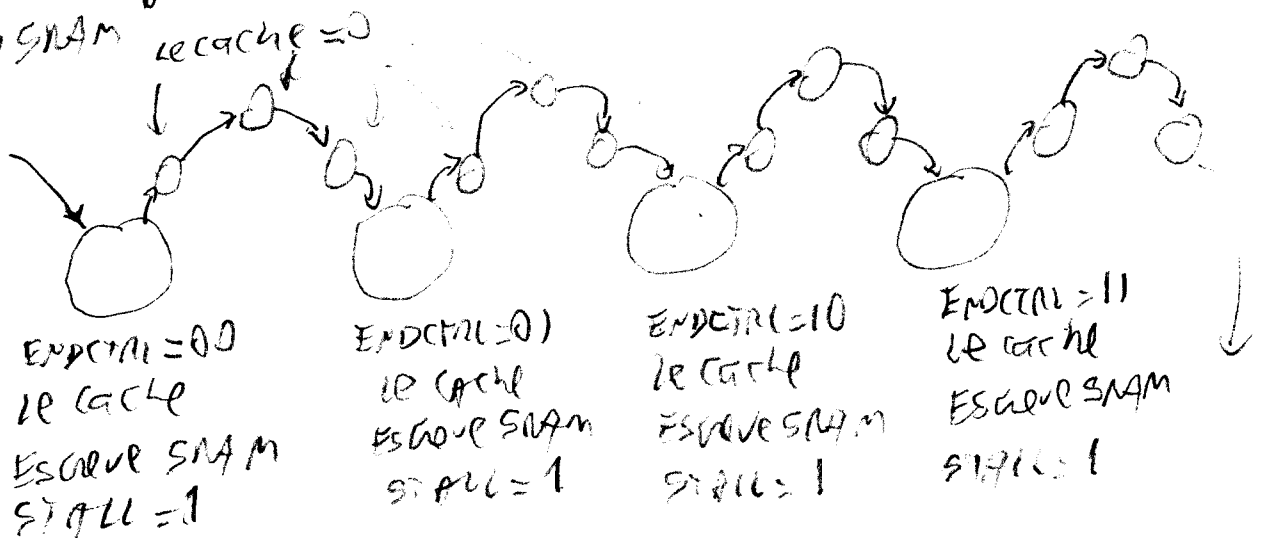
LE BLOCO  
SRAM → CACHE

0.7)



LE WORD  
SRAM → 4 CICLOS

ESCREVE BLOCO  
CACHE → SRAM



0.5)

LE  
WORD



ENDCTRL=00  
DE=X  
ESCREVE CACHE=0  
LE CACHE=1  
ESCREVE SRAM=0  
LE SRAM=0  
STALL=0

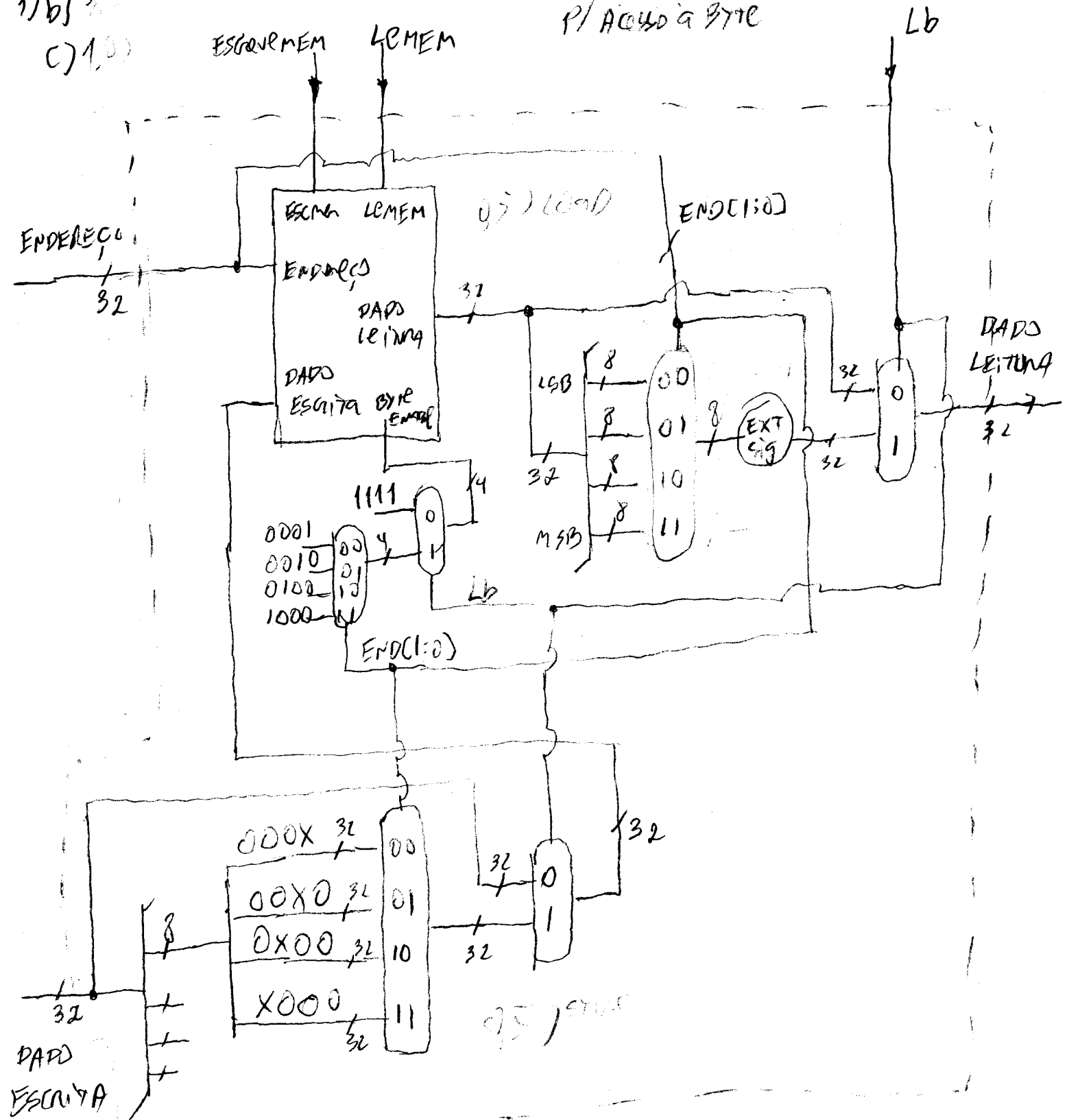
ESCREVE  
WORD



ENDCTRL=00  
DE=0  
ESCREVE CACHE=1  
LE CACHE=0  
ESCREVE SRAM=0  
LE SRAM=0  
STALL=0

1) b) 100  
 c) 100

# ESTRUTURA DA MEMÓRIA DE DADOS MODIFICADA P/ ACÚDO A BYTE



2)

a) UNICICLO: instrução mais lenta = SW CLEAR na cache (5)

$$T = 200p + 50p + 150p + 200p + 32 \cdot T_1 + 50p = 650p + 32 \cdot T_1 //$$

$$T_1 = \frac{1}{f} \text{ Frequência de acesso à SRAM}$$

$$f = \frac{1}{T} // (0,5)$$

b) MULTICICLO: ETAPA mais longa (considerando  $T_1 = 200ps + b_m$ )

$$T = 200ps \rightarrow f = 5GHz // (0,5)$$

$$N^{\circ} \text{ ciclos} = (5+16) + 5 + 4 + (4+32) + 4 + (5+16) + 5 + 4 + (4+16) + 3 + 3$$

$$= 126$$

$$t = 126 \times 200p = 25,2ns // (0,5)$$

c) PIPELINE 5 BOLHAS  $T = 200ps$ 

$$N^{\circ} \text{ ciclos} = 1+16+1+2+1+2+1+32+1+2+1+16+1+2+1+2+1+16+1+0+1+1 = 11+91 = 102$$

$$t = 102 \times 200p = 20,4ns // (0,5) \text{ bolha}$$

d) PIPELINE C/ FORWARD  $T = 200ps$ 

$$N^{\circ} \text{ ciclos} = 1+16+1+1+1+1+32+1+1+16+1+1+1+16+1+1+1 = 11+83 = 94$$

$$t = 94 \times 200p = 18,8ns // (0,5) \text{ bolha}$$