

## 5ª Lista de Exercícios

### Observação:

Os exercícios desta lista não serão cobrados. Porém, é altamente recomendável que os alunos tentem resolvê-los, a fim de se prepararem minimamente para a 3ª prova semestral.

### Exercício 1

Em diversas atividades econômicas há a necessidade de se realizar medidas de grandes distâncias, as quais seriam inviáveis sem o uso de equipamento dedicado, baseado em raio laser. O princípio de funcionamento de tal equipamento reside em emitir um “pulso curto” de raio laser em direção ao objeto de interesse (um prédio, uma montanha, uma ponte ou um anteparo, por exemplo) e registrar o tempo que transcorre desde que o raio foi disparado até que sua reflexão seja capturada por sensor ótico existente no próprio equipamento. Este princípio está ilustrado na fig. 1. Aproximando-se a velocidade do raio laser pela velocidade da luz, pode-se calcular a distância entre o equipamento medidor e o objeto-alvo (D, na equação) por meio da seguinte equação:  $2D = T[s] * 3 \times 10^8 [m/s]$ , onde T é o tempo transcorrido entre o disparo do laser e a captura do raio refletido.

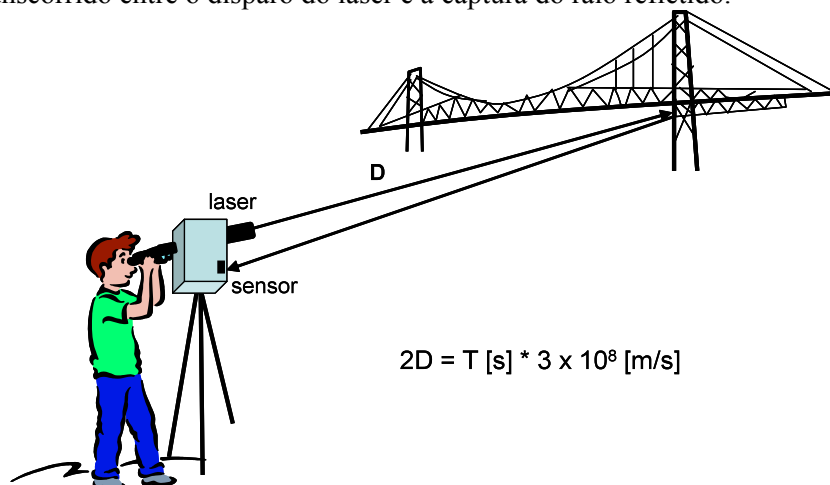


Fig. 1 - Medida de distância baseada no princípio da reflexão de raio laser.

Necessita-se de o sistema digital para controlar um equipamento capaz de medir distâncias de até 2.000 metros. Este sistema digital, doravante denominado de “**medidor**”, possui um sinal de entrada “B” que deve ser ativado para iniciar o processo de medida. Ao ser iniciado o processo de medida, o medidor ativa o sinal “L” que dispara o laser. A onda refletida é capturada por um sensor, o qual gera o sinal “S” (S=1 significa que foi captada onda refletida). O resultado da medição é mostrado em metros, codificado em binário pela saída “D”. A fig. 2 mostra as interfaces do medidor.

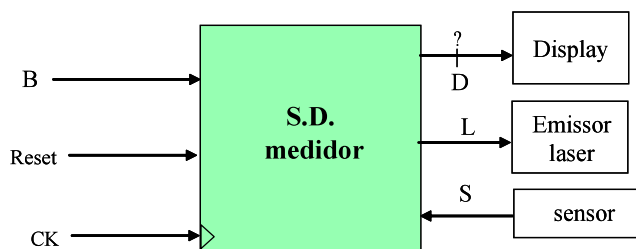


Fig. 2 - Interfaces do sistema digital “medidor”.

- a) Determine o número de bits para a variável D.

- Crie o diagrama de estados de alto nível (FSMD) para o sistema digital medidor. Assuma que existe um estado de “repouso” no qual o emissor laser não deve estar ativado e o display deve marcar distância= 0 (zero). Este estado de repouso é atingido por meio de um reset assíncrono ou quando o equipamento é ligado. Por outro lado, após ter sido feita uma medida, o equipamento deve manter no display o valor desta medida até que o cálculo de uma nova medida seja concluído. A medida é iniciada assim que o botão B é acionado (i.e., o sinal B é feito igual a “1”), o que causa a geração de um pulso de laser com duração igual a um ciclo de relógio. Assim que o sensor detectar o raio refletido, o medidor calcula a distância usando a equação. Enquanto o raio refletido não é detectado, um registrador vai sendo incrementado. Para que este registrador possa contar o tempo transcorrido, assuma que a frequência do relógio é de 300 MHz.
- Crie o B.O. (bloco operativo) para o sistema digital medidor e identifique todos os seus sinais de controle e de status.
- Assumindo o modelo de Moore, construa o diagrama de estados do bloco de controle (FSM).
- Construa a tabela de próximo estado e a tabela de sinais de saída.

## Exercício 2

Necessita-se de um sistema digital para calcular a média aritmética entre 4 pixels. As interfaces deste sistema digital, doravante denominado “Média4P”, são mostradas na Fig. 3. Os 4 pixels são disponibilizados nas entradas pixel1, pixel2, pixel3 e pixel4. “Média4P” possui um estado inicial “S0”, o qual corresponde ao estado de reset (assíncrono). Neste estado a saída “disponível” deve valer “1”. Quando o sinal de entrada “início” vale “1”, “Média4P” calcula a média aritmética entre os 4 pixels **em um único estado**, armazenando-a em um registrador de saída denominado “média”. Durante o cálculo, o sinal “disponível” deve valer “0”. As seguintes restrições devem ser observadas:

- Cada pixel é representado como um número inteiro sem sinal, com 8 bits.
- Os valores dos bits estão disponíveis e estáveis durante todo o tempo que for necessário.
- O bloco operativo deve ter custo mínimo, satisfeitas as especificações do enunciado da questão (sobretudo a que diz respeito ao cálculo da média ser realizado em um único ciclo de relógio).
- O número de estados deve ser mínimo.
- A divisão por 4 deve ser feita por meio de deslocamento de bits.

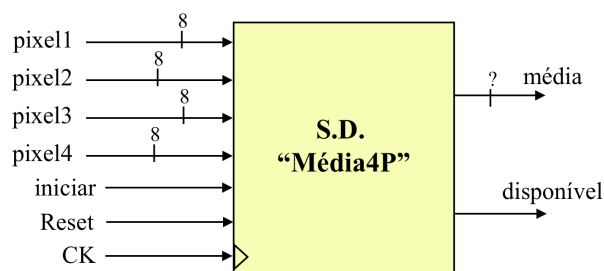


Fig. 3 - Interfaces do sistema digital “Média4P”.

- Determine o número de bits para a variável “média”.
- Desenhe o diagrama de estados de alto nível (FSMD) para “Média4P”.
- Desenhe o B.O. (bloco operativo) para “Média4P” e identifique todos os seus sinais de controle (comando e status).
- Assumindo os atrasos de componentes mostrados na Tab. 1, determine o período mínimo de relógio para controlar “Média4P”.
- Considerando somente os estados “úteis” (i.e., excluindo-se o estado de espera S0), estime o tempo de execução  $TE_{Média4P}$ .
- Assumindo o modelo de Moore, e baseando-se na FSMD e no B.O. projetado, desenhe a FSM do bloco de controle (FSM). Crie uma tabela de sinais de saída, anotando os valores de cada sinal de

comando para cada um dos estados. Crie a tabela de transição de estados. Encontre as equações simplificadas para os sinais de comando e para as variáveis de próximo estado.

g) Assumindo os valores de custos da Tab. 2, calcule o custo do B.O. de "Média4P"  $\text{custo}_{\text{Média4P}}$ .

Tab. 1 - Características temporais de componentes de B.O. para os exercícios desta lista.

Componente	característica	símbolo	valor
Memória com 4 linhas	Tempo para leitura de uma linha	$t_{\text{leit}}$	40 ps
Registradores	Tempo de setup	$t_{\text{SU}}$	4 ps
Registradores	Tempo de hold	$t_{\text{H}}$	Desprezível (0 ps)
Registradores	Tempo de carga	$t_{\text{CO}}$	5 ps
Mux 2:1	Atraso crítico	$t_{\text{DMUX}}$	2 ps
Somador completo ( <i>full adder</i> )	Atraso crítico	$t_{\text{DS}}$	10 ps
Registrador de estados (do controle)	tempos de setup, carga e hold	-	Desprezível (0 ps)
Lógica de saída do controle	atraso	-	Desprezível (0 ps)

Tab. 2 - Custos de componentes de B.O. para os exercícios desta lista.

Componente do Nível RT	Custo
Somador	$24 \times n$
Subtrator	$26 \times n$
Somador/subtrator	$30 \times n$
Multiplicador	$30 \times n^2$
Multiplexador 2:1	$4 \times n$
Multiplexador 3:1	$8 \times n$
Multiplexador 4:1	$12 \times n$
Registrador com carga paralela (+4n para set ou reset assíncrono)	$18 \times n$
Registrador com carga paralela controlada (+4n para set ou reset assíncrono)	$22 \times n$
Registrador de deslocamento (+4n para set ou reset assíncrono)	$18 \times n$
Registrador de deslocamento com carga paralela controlada (+4n para set ou reset assíncrono)	$22 \times n$
Buffer Tri-state inversor	$4 \times n$
Buffer Tri-state não-inversor	$6 \times n$
Comparador entre dois números, possivelmente diferentes de zero	$6 \times n$
Fios e demais componentes não citados (incluindo o comparador para zero)	desprezível

### Exercício 3

Este exercício refere-se a uma segunda versão de um sistema digital capaz de calcular a média aritmética entre 4 pixels, doravante denominado "Média4Pv2". As interfaces "Média4Pv2" são mostradas na Fig. 4. Os 4 pixels estão armazenados em uma memória externa que possui 4 linhas de 8 bits cada. "Média4Pv2" possui um estado inicial "S0", o qual corresponde ao estado de reset (assíncrono). Neste estado a saída "disponível" deve valer "1". Quando o sinal de entrada "início" vale "1", "Média4Pv2" inicia o cálculo da média, etapa na qual o sinal "disponível" deve valer "0". Para a realização do cálculo da média os pixels precisam ser lidos da memória (um a um) e somados. Ao final, o resultado da média é armazenado em um registrador de saída denominado "média". As seguintes restrições devem ser observadas:

- Cada pixel é representado como um número inteiro sem sinal, com 8 bits. Assim, cada linha da memória da Fig. 4 armazena somente um pixel.
- Para realizar a leitura da memória, o sinal "lê" deve valer 1 e o sinal "end" deve conter o endereço da linha a ser lida (0, 1, 2 ou 3, em binário). A leitura de uma linha da memória requer que os sinais "lê" e "end" fiquem estáveis durante o tempo necessário para a leitura de uma linha (conforme Tab. 1), ao final do qual o pixel armazenado estará disponível na saída da memória (sinal "pixel").
- O bloco operativo deve ter custo mínimo, satisfeitas as especificações do enunciado da questão.
- O número de estados deve ser mínimo.
- A divisão por 4 deve ser feita por meio de deslocamento de bits.

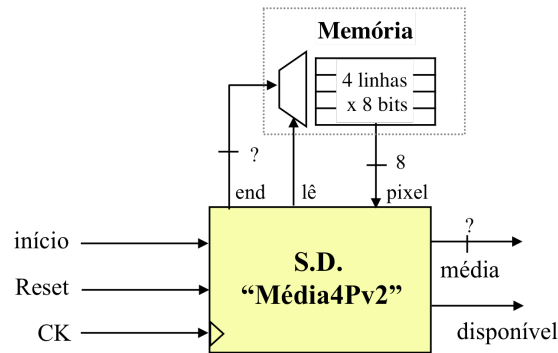


Fig. 4 - Interfaces do sistema digital “Média4Pv2”.

- Determine o número de bits para a variável “end”.
- Desenhe o diagrama de estados de alto nível (FSMD) para “Média4Pv2”.
- Desenhe o B.O. (bloco operativo) para “Média4Pv2” e identifique todos os seus sinais de controle (comando e status).
- Assumindo os atrasos de componentes mostrados na Tab. 1, determine o período mínimo de relógio para controlar “Média4Pv2”.
- Considerando somente os estados “úteis” (i.e., excluindo-se o estado de espera S0), estime o tempo de execução  $TE_{Média4Pv2}$ .
- Calcule a relação  $TE_{Média4Pv2}/TE_{Média4P}$  e comente o resultado obtido.
- Assumindo o modelo de Moore, e baseando-se na FSMD e no B.O. projetado, desenhe a FSM do bloco de controle (FSM). Crie uma tabela de sinais de saída, anotando os valores de cada sinal de comando para cada um dos estados. Crie a tabela de transição de estados. Encontre as equações simplificadas para os sinais de comando e para as variáveis de próximo estado.
- Assumindo os valores de custos da Tab. 2, calcule o custo do B.O. de “Média4Pv2”,  $custo_{Média4Pv2}$ .
- Calcule a relação  $custo_{Média4Pv2}/custo_{Média4P}$  e comente o resultado obtido. Comente também a relação entre o custo (em termos de sinais de comando e variáveis de estados) do B.C. (bloco de controle) de “Média4Pv2” e o custo do B.C. de “Média4P”.

## Exercício 4

Necessita-se de um sistema digital capaz de calcular a média aritmética das 4 medições mais recentes de temperatura, provenientes de um sensor de temperatura externo. O sensor externo mede constantemente a temperatura. Entretanto, o sistema digital, doravante denominado “Média\_Temp”, amostra o valor de temperatura apenas na borda de subida do sinal de relógio. Enquanto o sinal de início estiver ativo (início = 1), a cada novo ciclo de relógio “Média\_Temp” amostra um novo valor de temperatura, calcula uma nova média aritmética (utilizando o novo valor e os três valores mais recentes) **em um único ciclo de relógio** e armazena esta nova média em um registrador de saída no ciclo de relógio seguinte. O conteúdo deste registrador de saída corresponde à saída “média” de “Média\_Temp”. O sinal de saída “válido” só deve valer “1” quando o valor disponível na saída “média” corresponder à média aritmética entre quatro medições de temperatura. (Isto significa que, entre o ciclo de relógio em que “início” começa a valer “1” e o ciclo de relógio em que é disponibilizado na saída “média” o valor da média entre as primeiras quatro temperaturas amostradas, “válido” deve valer “0”.) Considere que as temperaturas amostradas estão dentro do intervalo  $[-250^{\circ}\text{C}; +250^{\circ}\text{C}]$ . A exemplo do exercício anterior, a divisão por 4 deve ser feita por meio de deslocamento de bits. A figura 5 ilustra os sinais de interface de Média\_Temp.

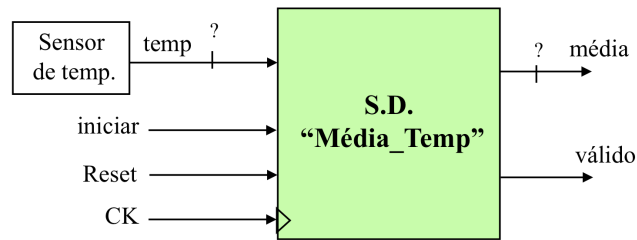


Fig. 5 - interfaces do sistema digital “Média\_Temp”.

- Determine o número de bits para as variáveis "temp" e “média”.
- Desenhe o diagrama de estados de alto nível (FSMD) para “Média\_Temp”.
- Desenhe o bloco operativo para “Média\_Temp” e identifique todos os seus sinais de controle (comando e status).
- Assumindo o modelo de Moore, e baseando-se na FSMD e no B.O. projetado, desenhe a FSM do bloco de controle (FSM). Crie uma tabela de sinais de saída, anotando os valores de cada sinal de comando para cada um dos estados. Crie a tabela de transição de estados. Encontre as equações simplificadas para os sinais de comando e para as variáveis de próximo estado.

## Exercício 5

Considere o “Exemplo 3” de bloco acelerador visto na aula 12T, o qual realiza o cálculo da SAD (*Sum of Absolute Differences*) para duas matrizes de 8x8 pixels, considerando que cada pixel está representado como um número binário de 8 bits. A Fig. 6 mostra a FSMD para a “versão 1” deste bloco acelerador, ao passo que a Fig. 7 mostra o bloco operativo (B.O.) para a “versão 2. Ambas versões assumem que as matrizes que armazenam os 64 pixels estão organizadas em 64 linhas de 8 bits (64 x 8). Já a Fig. 8 mostra a FSMD da “versão 3”, também vista em aula, a qual assume duas memórias estão organizadas em 16 linhas de 32 bits (16 x 32).

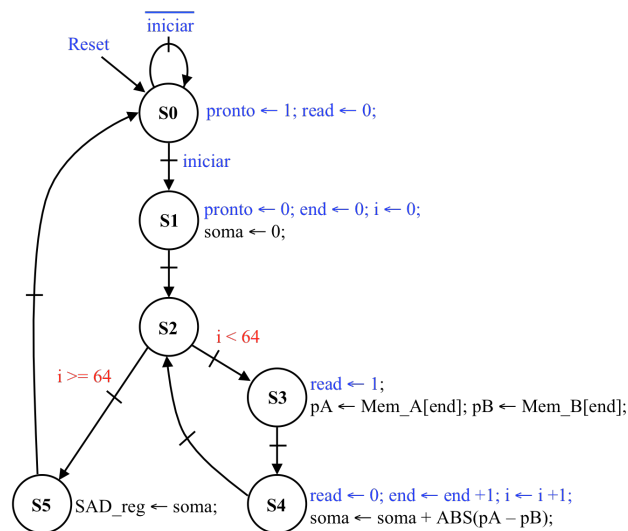


Fig. 6 - FSMD da “versão 1” do bloco acelerador que calcula a SAD para duas matrizes de 8x8 pixels.

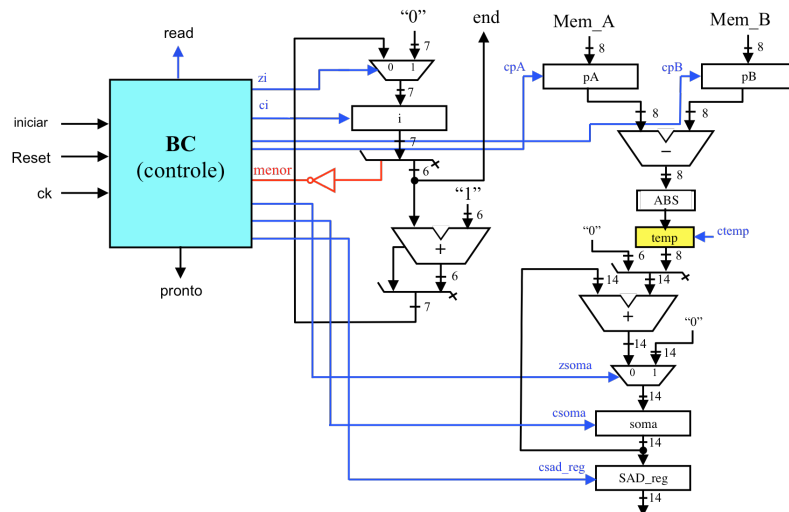


Fig. 7 - B.O. da “versão 2” do bloco acelerador que calcula a SAD para duas matrizes de 8x8 pixels.

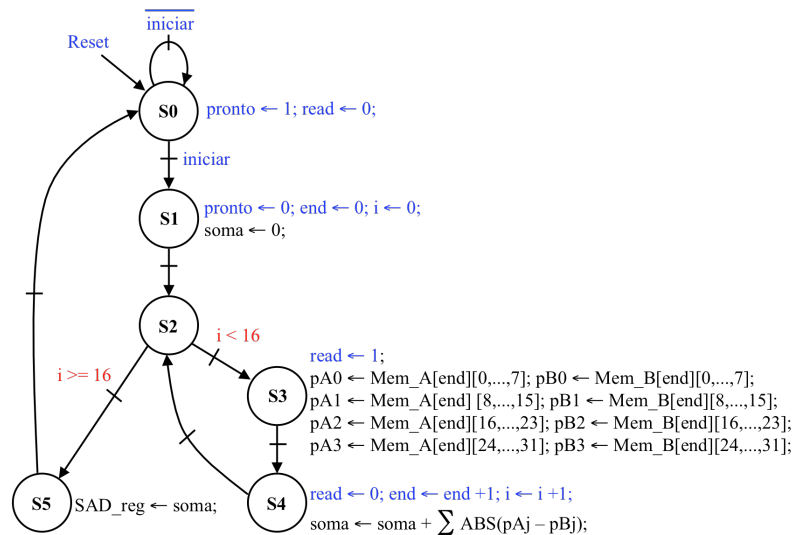


Fig. 8 - FSMD da “versão 3” do bloco acelerador que calcula a SAD para duas matrizes de 8x8 pixels.

- Desenhe a FSMD para a “versão 2” do bloco acelerador da SAD;
- Usando os dados da Tab. 2, calcule os custos dos B.O.s das três versões de bloco acelerador da SAD. Assuma que a unidade funcional que calcula o módulo de um número (“ABS”) tem o mesmo custo de um subtrator;
- Comente os custos dos B.O.s e dos B.C.s destas três versões de bloco acelerador de SAD, estabelecendo um paralelo com os respectivos tempos de execução atingidos.
- Quais seriam as implicações de se adiantar a leitura das memórias para o estado S2. Comente para cada uma das três versões. Observe que o estado S2 sempre é realizado uma vez mais do que qualquer um dos demais estados do laço.

## Exercício 6

- Desenhe uma nova versão de B.O. (bloco operativo) para o bloco acelerador do exemplo 4, o qual seja capaz de fornecer o resultado com máxima precisão, i.e., evitando a ocorrência de *overflow*;
- Usando os dados da Tab. 1, calcule o período mínimo do relógio e o tempo de execução para esta nova versão de bloco acelerador. Aproxime o atraso do subtrator e do somador/subtrator pelo atraso do somador.
- Usando os dados da Tab. 2, calcule o custo desta nova versão de B.O.

### Exercício 7

- Desenhe uma nova versão de B.O. (bloco operativo) para o bloco acelerador do exemplo 5, o qual seja capaz de fornecer o resultado com máxima precisão, i.e., evitando a ocorrência de *overflow*;
- Usando os dados da Tab. 1, calcule o período mínimo do relógio e o tempo de execução para esta nova versão de bloco acelerador. Aproxime o atraso do subtrator e do somador/subtrator pelo atraso do somador.
- Usando os dados da Tab. 2, calcule o custo desta nova versão de B.O.

### Exercício 8

- Compare os tempos de execução das versões de bloco acelerador dos exercícios 6 e 7 com o tempo de execução do bloco acelerador do exemplo 6, conforme calculado em aula;
- Compare os custos das versões de bloco acelerador dos exercícios 6 e 7 com o custo do bloco acelerador do exemplo 6, conforme calculado em aula;

### Exercício 9

Seja o Algoritmo 1, apresentado a seguir.

```
Algoritmo 1: divisão inteira
1 Início Divide
2   dividendo ← ent1; divisor ← ent2; res ← 0;
3   Enquanto dividendo >= divisor faça
4     início
5       res ← res + 1;
6       dividendo ← dividendo - divisor;
7     fim
8   saída1 ← res; saída2 ← dividendo;
9 fim
```

Necessita-se de um bloco acelerador com **B.O. de custo mínimo** para executar este algoritmo, doravante referenciado por “Divisor-v1”. Este bloco possui duas entradas de dados, “ent1” e “ent2”, pelas quais são fornecidos (simultaneamente) dois operandos inteiros positivos com sinal (e portanto, bit de sinal=0), representados com **8 bits**. Os resultados do cálculo, “res” e “dividendo”, devem ficar disponíveis mesmo durante a execução de um novo cálculo. **Não há restrições quanto ao desempenho de “Divisor-v1”**. Além das entradas de dados, “Divisor-v1” também possui uma entrada de controle denominada “iniciar”, a qual comanda o início de um novo cálculo, e uma saída de controle denominada “pronto”, a qual sinaliza o término de um cálculo. Também há uma entrada de “Reset” assíncrono, a qual coloca a máquina de estados no estado de espera, S0 (como de praxe nesta disciplina).

- Desenhe a FSMD para “Divisor-v1”;
- Desenhe o B.O. de “Divisor-v1”;
- Desenhe a FSM de “Divisor-v1”;
- Assumindo o modelo de Moore, monte a tabela de sinais de saída e a tabela de transição de estados para “Divisor-v1”. Identifique o número de sinais de saída diferentes entre si;
- Usando a Tab. 1, calcule o período mínimo do relógio e o tempo de execução de “Divisor-v1”;
- Usando a Tab. 2, calcule o custo do B.O. de “Divisor-v1”.

### Exercício 10

Considere ainda o Algoritmo 1, apresentado no exercício 9. Necessita-se de um bloco acelerador de **máximo desempenho** para executar este algoritmo, doravante referenciado por “Divisor-v2”. Este

bloco possui duas entradas de dados, “ent1” e “ent2”, pelas quais são fornecidos (simultaneamente) dois operandos inteiros sem sinal, representados com **8 bits**. Os resultados do cálculo, “res” e “dividendo”, devem ficar disponíveis mesmo durante a execução de um novo cálculo. **Não há restrições quanto ao custo do B.O. de “Divisor-v2”**. Além das entradas de dados, “Divisor-v2” também possui uma entrada de controle denominada “iniciar”, a qual comanda o início de um novo cálculo, e uma saída de controle denominada “pronto”, a qual sinaliza o término de um cálculo. Também há uma entrada de “Reset” assíncrono, a qual coloca a máquina de estados no estado de espera, S0 (como de praxe nesta disciplina).

- Desenhe a FSMD para “Divisor-v2”;
- Desenhe o B.O. de “Divisor-v2”;
- Desenhe a FSM de “Divisor-v2”;
- Assumindo o modelo de Moore, monte a tabela de sinais de saída e a tabela de transição de estados para “Divisor-v2”. Identifique o número de sinais de saída diferentes entre si;
- Usando a Tab. 1, calcule o período mínimo do relógio e o tempo de execução de “Divisor-v2”;
- Usando a Tab. 2, calcule o custo do B.O. de “Divisor-v2”.

### Exercício 11

O diagrama da Fig. 9a mostra as interfaces do sistema digital “**Médiav1**” com o mundo externo e com uma memória, referenciada por “Mem”. **Médiav1** deve calcular a média aritmética (valor inteiro, truncado) entre **64** valores inteiros sem sinal (representados em binário, com 8 bits cada) que estão armazenados em **Mem**. Este cálculo deve ser feito com máxima precisão, mas o resultado deve ser truncado para 8 bits.

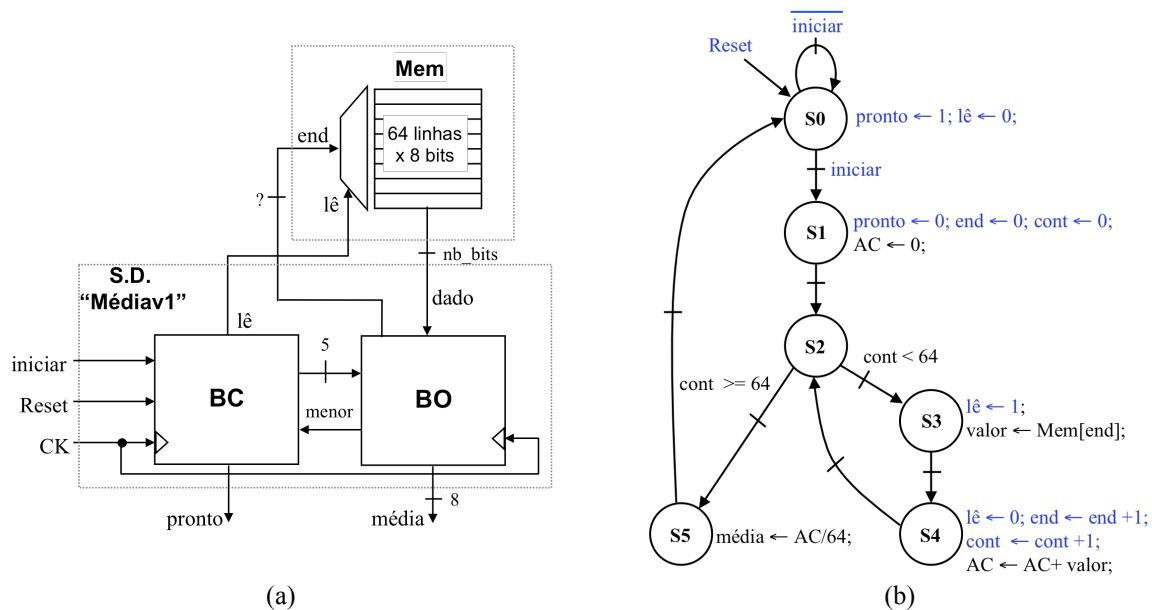


Fig. 9 – Interfaces do sistema digital **Médiav1** (a) e sua FSMD (b).

A Fig. 9b mostra a FSMD (diagrama de estados de alto nível), que representa o comportamento de **Médiav1**. Nesta FSMD, “Mem[**end**]” representa um acesso à linha de “Mem” cujo endereço é “**end**”. Além disso, esta FSMD assume que:

- “Mem” está organizada como uma matriz de 64 linhas de 8 bits. Antes que “**iniciar**” = 1, “Mem” foi devidamente carregada com os 64 valores binários inteiros, sem sinal, de 8 bits.
- A leitura de uma linha de “Mem” é assíncrona: deve-se manter “**end**” estável e o sinal **lê**=1 durante 1,8 ns. Após este tempo (referenciado por **tleit** na Tab. 4), o valor armazenado “Mem[**end**]” estará disponível na saída “**dado**” de “Mem”, a qual também é entrada de **Médiav1**.



- O período do relógio CK deverá ser suficientemente longo para acomodar as operações assinaladas a qualquer um dos estados da FSM (inclusive a leitura de "Mem".)

A Fig. 10 mostra o diagrama incompleto do bloco operativo (B.O.) de **Médiav1**. Para responder os itens "a" a "i" deste exercício, assuma as seguintes especificações com relação ao B.O. da Fig. 10:

- O registrador "cont" será usado para implementar ambas variáveis "end" e "cont" da FSM da Fig. 9b (semelhantemente ao visto em aula). Este registrador possui carga paralela ativada pelo sinal "ccont";
- Deve haver um somador a ser usado exclusivamente para o incremento de "cont";
- Os registradores "valor", "AC" e "média" possuem carga paralela, ativada pelos sinais "cvalor", "cAC" e "cmédia", respectivamente;
- A divisão por "64" (estado S5, na Fig. 9b) é inteira. Seu resultado deve ser truncado para 8 bits;
- Satisfeitas as especificações anteriores, o custo do bloco operativo de **Médiav1** deve ser **mínimo**.

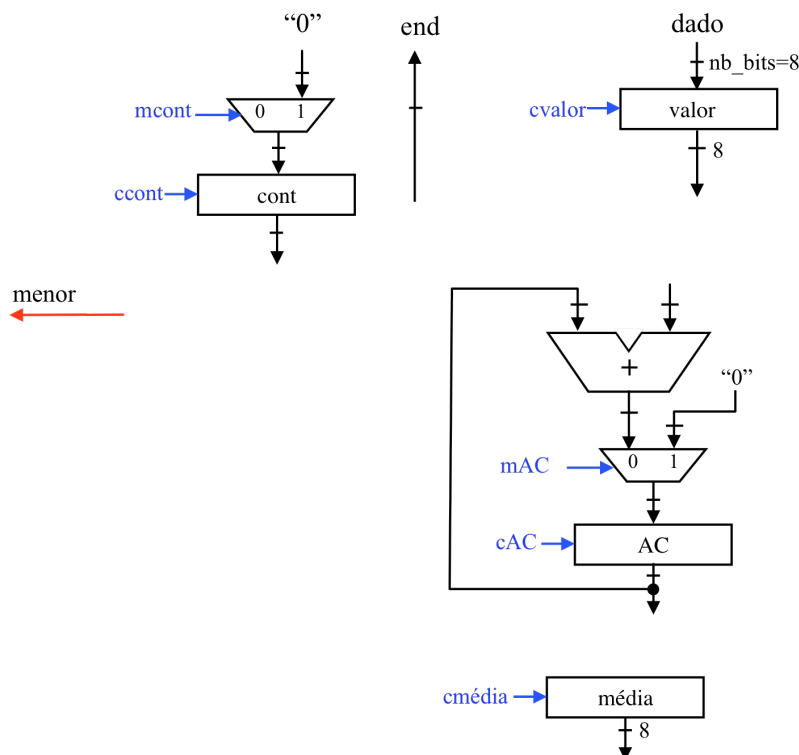


Fig. 10 – Diagrama do bloco operativo (B.O.) de **Médiav1** (desenho a completar).

- Quantos bits deve ter o contador "cont" do B.O. de **Médiav1**? Justifique sua resposta mostrando os cálculos.
- Quantos bits deve ter a saída "end" do B.O. de **Médiav1**? Justifique sua resposta mostrando os cálculos.
- Quantos bits deve ter o somador que aparece no B.O. de **Médiav1**? Justifique sua resposta mostrando os cálculos.

Com base nas informações das Figs. 9a e 9b, e respeitando as especificações fornecidas, complete no diagrama do B.O. de **Médiav1** mostrado na Fig. 10:

- Os componentes e as conexões (de dados) para "cont" gerar as saídas "end" e "menor".
  - Os componentes e as conexões (de dados) entre "valor" e o somador.
  - As conexões entre "AC" e "média".
- Calcule o número de ciclos de relógio necessários para que **Médiav1** realize o cálculo da média aritmética entre os **64** valores armazenados em "Mem". Mostre a sequência de estados para este cálculo. Desconsidere o estado S0.

- e) Levando em consideração a FSMD da Fig. 9b, o B.O. da Fig. 10 e o enunciado desta questão, complete na Tab. 3 os valores adequados para os sinais de comando que controlam o B.O. de **Médiav1**.

Tab. 3 – Valores dos sinais de comando que controlam o B.O. de “Médiav1” (a completar).

Estado	mcont	ccont	lê	mvalor	mAC	cAC	cmédia
S0							
S1							
S2							
S3							
S4							
S5							

- f) Desenhe a FSM para **Médiav1**.

A Tab. 4 mostra as características temporais dos componentes de **Médiav1**. Baseando-se nestas características, na FSMD da Fig 9b, no enunciado desta questão e no B.O. da Fig. 10, responda o que é solicitado nos itens “g”, “h” e “i”.

Tab. 4 – Características temporais dos componentes usados na implementação de “Médiav1”.

Componente	Característica	Símbolo	Valor
Memória ("Mem") de 64 linhas x 8 bits	tempo para leitura de uma linha	tleit64	1,8 ns
Memória ("Mem") de 16 linhas x 32 bits	tempo para leitura de uma linha	tleit16	1,0 ns
Somador completo ( <i>full adder</i> )	atraso crítico	tds	0,1 ns
Registradores	tempo de setup	tsu	0,2 ns
Registradores	tempo de carga	tco	0,3 ns
Registradores	tempo de hold	th	Desprezível (0 ns)
Mux 2:1	atraso	tdmux	0,3 ns
Sinais de controle (qualquer um) *	atraso	d	Desprezível (0 ns)
Outras operações não especificadas acima	atraso	-	Desprezível (0 ns)

\* Isto equivale a assumir que tco, th e tsu do registrador de estados é zero e que o atraso da lógica de saída também é zero.

- g) Calcule o tempo (atraso) para executar cada um dos estados (**S0, S1, S2, S3, S4 e S5**) de **Médiav1**. Para cada cálculo, indique as parcelas de atraso e seus respectivos valores.
- h) Considerando os tempos calculados no item anterior, determine a frequência máxima do relógio para **Médiav1**, expressando-a em MHz (megahertz).
- i) Utilizando os resultados dos itens anteriores, calcule o tempo total para que **Médiav1** realize o cálculo da média aritmética entre os **64** valores armazenados em "Mem". Neste cálculo, desconsidere as possíveis execuções de S0.

Suponha que se deseje criar uma nova versão de **Médiav1**, denominada **Médiav2**, na qual a memória é organizada como uma matriz de 16 linhas de 32 bits cada. Desta forma, em cada linha são armazenados 4 valores e portanto, cada acesso de leitura traz 4 valores de uma só vez. Esta nova organização de “Mem” deve ser explorada, de modo a permitir um aumento de desempenho de **Médiav2** em relação a **Médiav1**, o que exigirá a adaptação da FSMD da Fig. 3b. Por outro lado, o número de bits dos somadores a serem utilizados no B.O. de **Médiav2** deve ser minimizado. Baseando-se nestas especificações para **Médiav2** (e também nas especificações de **Médiav1**), e nos dados da Tab. 4, responda aos itens “j” a “p”.

- j) Respeitando as especificações acima, complete no diagrama do B.O. de **Médiav2** mostrado na Fig. 11. Não esqueça de identificar o número de bits em cada conexão!

- Os componentes e as conexões (de dados) para "cont" gerar as saídas "end" e "menor".

- Os componentes e as conexões (de dados) entre os registradores “valor0”, “valor1”, “valor2”, “valor3” e o somador.
- As conexões entre “AC” e “média”.

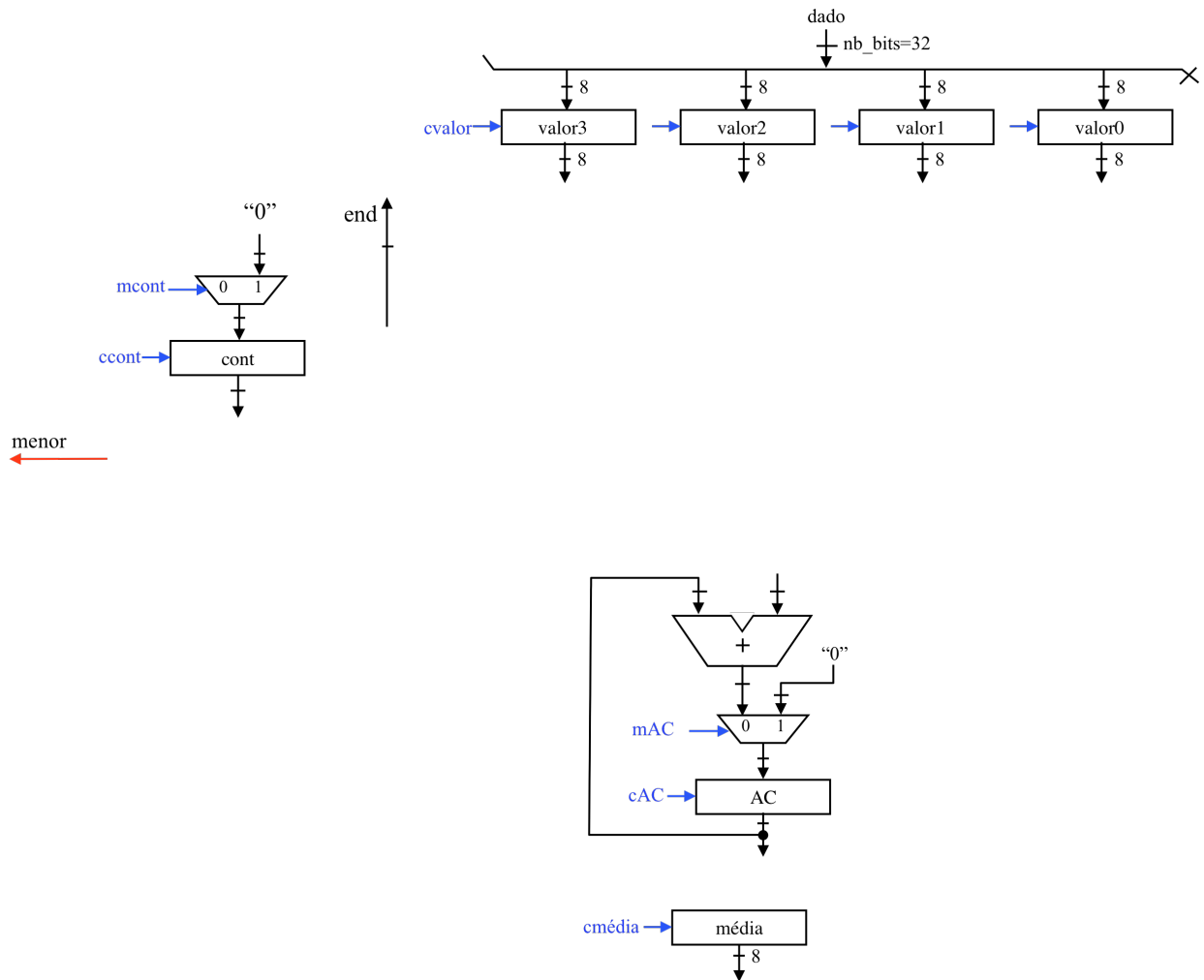
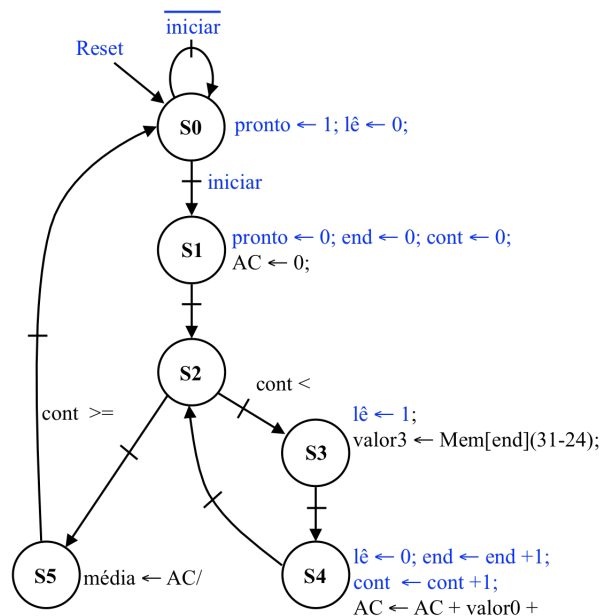


Fig. 11 – Diagrama do bloco operativo (B.O.) de **Médiav2** (desenho a completar).

k) Complete a FSMD para **Médiav2**, mostrada na Fig. 12.

Fig. 12 – FSMD de **Média2** (desenho a completar).

- Calcule o número de ciclos de relógio necessários para que **Média2** realize o cálculo da média aritmética entre os **64** valores armazenados em "Mem". Mostre a sequência de estados para este cálculo. Desconsidere o estado S0.
- Quais modificações precisam ser feitas na Tab. 3 para que esta contenha os sinais de comando para Média2?
- Calcule o tempo (atraso) para executar cada um dos estados (S0, S1, S2, S3, S4 e S5) de Média2. Para cada cálculo, indique as parcelas de atraso e seus respectivos valores.
- Considerando os tempos calculados no item anterior, determine a frequência máxima do relógio para Média2, expressando-a em MHz (megahertz).
- Utilizando os resultados dos itens anteriores, calcule o tempo total de execução do algoritmo realizado por Média2. Desconsidere as possíveis execuções de S0.

Para os itens seguintes deste exercício considere os custos dos componentes dos blocos operativos de Média1 e de Média2 conforme indicado na Tab. 5.

Tab. 5 – Custos (em número de transistores) dos componentes dos B.O.s de Média1 e de Média2.

Componente	Custo para $n$ bits
Somador de $n$ bits	$24n$
Registradores com carga paralela controlada	$22n$
Mux 2:1	$4n$

- Calcule a relação custo\_BO\_Média2/custo\_BO\_Média1, isto é, a relação entre o custo do B.O. de Média1 e o custo do B.O. de Média2. Neste cálculo, ignore os custos das respectivas memórias.
- Utilizando os resultados dos itens "l" e "d", calcule a relação  $N_{\text{ciclos\_Média2}}/N_{\text{ciclos\_Média1}}$ , onde  $N_{\text{ciclos\_Média2}}$  e  $N_{\text{ciclos\_Média1}}$  correspondem, respectivamente, ao número de ciclos de relógio necessários para que Média2 e Média1 realizem o cálculo da média aritmética entre os 64 valores armazenados em suas respectivas memórias.
- Utilizando os resultados dos itens "o" e "h", calcule a relação  $T_{\text{Média2}}/T_{\text{Média1}}$ , onde  $T_{\text{Média2}}$  e  $T_{\text{Média1}}$  correspondem, respectivamente, ao período dos relógios a serem aplicados

em **Médiav2** e **Médiav1**.

- t) Utilizando os resultados dos itens "p" e "i", calcule a relação  $\text{tempo\_Médiav2} / \text{tempo\_Médiav1}$ , onde  $\text{tempo\_Médiav2}$  e  $\text{tempo\_Médiav1}$  correspondem, respectivamente, aos tempos necessários para que **Médiav2** e **Médiav1** realizem o cálculo da média aritmética entre os 64 valores armazenados em suas respectivas memórias.
- u) Comente de maneira crítica os resultados obtidos nos itens "q", "r", "s" e "t", de modo a resumir todos os itens anteriores (i.e., resumindo a exploração do espaço de projeto realizada neste exercício).