



# Universidade Federal de Santa Catarina

Departamento de Informática e Estatística  
Bacharelado em Ciências da Computação

INE 5348 – Lógica Programável – semestre 2007/2

Prof. José Luís Güntzel guntzel@inf.ufsc.br www.inf.ufsc.br/~guntzel

## Lista de Exercícios Preparatórios para a 2ª Verificação

Para os exercícios a seguir, considere a seguinte tabela de custo de operadores:

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 (+2n para set ou reset assíncrono)	$18 \times n$
Registrador com carga paralela controlada (+2n para set ou reset assíncrono)	$22 \times n$
Buffer Tri-state inversor	$4 \times n$
Buffer Tri-state não-inversor	$6 \times n$
Fios e demais componentes (incluindo o comparador)	desprezível

Custo dado em número aproximado de transistores, onde  $n$  = número de bits do componente

### Exercício 1

Considere o algoritmo descrito a seguir. Assuma que todas as variáveis são inteiras sem sinal, com  $n$  bits.

```
1  Início Diffeq
2  Leia x, y, u, dx, a;
3  Repete
4      x1 ← x + dx;
5      u1 ← u - (3*x*u*dx) - (3*y*dx);
6      y1 ← y + u*dx;
7      x ← x1; u ← u1; y ← y1;
8  Até (x1 >= a)
9      saída ← y;
10 fim
```

Considere que, para reduzir o custo do encapsulamento do circuito, existem apenas 2 entradas (cada uma com  $n$  bits), ent1 e ent2, pelas quais devem ser lidas as variáveis de entrada  $x$ ,  $y$ ,  $u$ ,  $dx$  e  $a$ .

a) Quantos ciclos de relógio serão necessários para a leitura das variáveis de entrada? Comente a leitura das variáveis de entrada.

Considere a possibilidade de se utilizar os seguintes tipos de UFs (unidades funcionais), capazes de operar sobre dois operandos por vez: somadores, subtratores, somadores/subtratores, multiplicadores

Considerando apenas o trecho de código compreendido entre as linhas 4 a 6 do algoritmo dado, estime o **custo de implementação** do bloco operativo e o respectivo **tempo de execução** (em ciclos de relógio) para as situações listadas a seguir. Faça uso de Grafos de Fluxo de Dados (DFG – *Data flow Graphs*). Para a estimativa de custo, considere apenas os custos das UFs necessárias (desconsidere registradores e rede de interconexão).

b) Mínimo custo para o máximo desempenho (mínimo tempo de execução) que este algoritmo pode atingir.

c) Máximo desempenho, assumindo o uso de um multiplicador e um somador/subtrator.

- d) Desenhe o fluxograma de execução deste algoritmo para o caso do item “c”.
- e) Determine o número mínimo de registradores necessários para a versão do item “e”. Faça a tabela de tempo de vida das variáveis e explique o resultado.
- f) Realize o assinalamento de variáveis a registradores segundo o algoritmo *left-edge*.
- g) Usando o resultado do item anterior e assumindo o uso de fios e multiplexadores, desenhe o bloco operativo no nível RT para a versão do algoritmo descrito no item “e”.
- h) Assumindo o modelo de Moore, desenhe o diagrama de estados para o controle do bloco operativo do item anterior.
- i) Faça uma estimativa do custo do bloco operativo do item “h” e do bloco de controle associado ao item “i” (para este último, apenas o número de estados e de sinais de saída). Comente o impacto do paralelismo atingido no bloco operativo sobre o custo do bloco de controle.
- j) Refaça os itens “e” a “j”, considerando a execução do algoritmo desta questão segundo o caso do item “c”.
- k) Proponha uma versão *pipeline* para este algoritmo.

## Exercício 2

Para o algoritmo abaixo, considere que os valores fornecidos pelas entradas “ent1” e “ent2” são inteiros sem sinal representados com  $n$  bits.

Para realizar o teste “divisor < dividendo”, assumo o uso de um comparador (em todas as implementações consideradas), com custo igual ao de um subtrator.

```

1  Início Divide
2      dividendo ← ent1; divisor ← ent2; res ← 0;
3      Enquanto divisor < dividendo faça
4          início
5              res ← res + 1;
6              dividendo ← dividendo - divisor;
7          fim
8      saída1 ← res; saída2 ← dividendo;
9  fim

```

Para responder os itens “a” até “c” abaixo, considere uma implementação “serial de custo mínimo” (sem uso de *pipeline* e sem exploração de paralelismo nas operações), na qual os passos 3 a 7 são realizados um após o outro (necessariamente).

- a) Desenhe o fluxograma e o diagrama de estados para a versão “serial de custo mínimo”.
- b) Qual é o menor número de registradores que se pode utilizar? Faça a tabela de tempo de vida das variáveis e explique o resultado.
- c) Desenhe o bloco operativo no nível RT para a versão “serial de custo mínimo”.
- d) Desenhe o fluxograma e o diagrama de estados para a versão “paralela”.
- e) Qual é o menor número de registradores que se pode utilizar? Faça a tabela de tempo de vida das variáveis e explique o resultado.
- f) Desenhe o bloco operativo no nível RT para a versão “paralela”.
- g) Compare a versão “serial de custo mínimo” com a versão “paralela”, em termos de custo (calcule os custos de ambas) e de desempenho (analise o melhor e pior caso para cada versão).
- h) Suponha que se deseje utilizar um subtrator para o teste “divisor < dividendo”. Qual seria o impacto desta decisão nos itens anteriores?