



Universidade Federal de Santa Catarina

Centro Tecnológico

Departamento de Informática e Estatística

Curso de Graduação em Ciências da Computação



Lógica Programável

INE 5348

Aula 11

Projeto de Sistemas Digitais no Nível RT. Estudos de caso. Projeto do multiplicador por somas sucessivas (sol.2- máx. desempenho). Projeto do multiplicador por somas sucessivas (sol.3). Multiplicadores Combinacionais.

Prof. José Luís Güntzel

guntzel@inf.ufsc.br

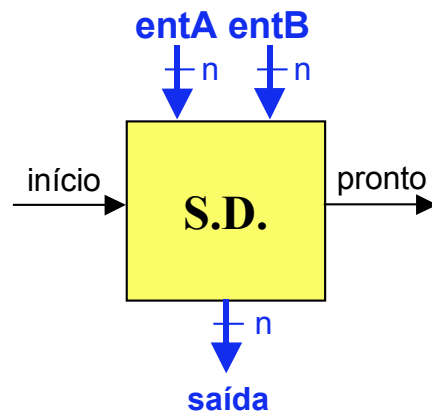
www.inf.ufsc.br/~guntzel/ine5348/ine5348.html

Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO Visando Máximo Desempenho

Exemplo 1: Projetar um BO para o SD que implementa o algoritmo abaixo, assumindo que:

- O SD possua duas entradas de dados
- **O SD precisa ter alto desempenho**
- Não há restrição quanto ao custo

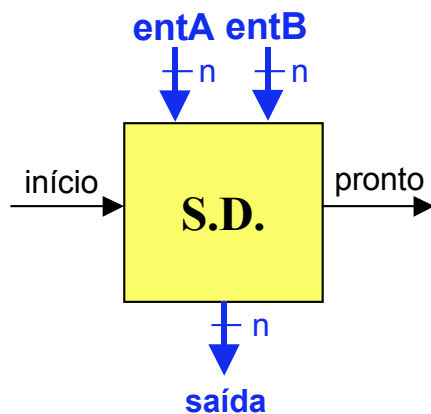


```
início
pronto ← 0;
A ← entA;
B ← entB;
P ← 0;
Se B ≠ 0 então
Enquanto A ≠ 0 faça
  início
  P ← P + B;
  A ← A - 1;
  fim
saída ← P;
pronto ← 1;
fim
```

Projeto de Sistemas Digitais no Nível RT

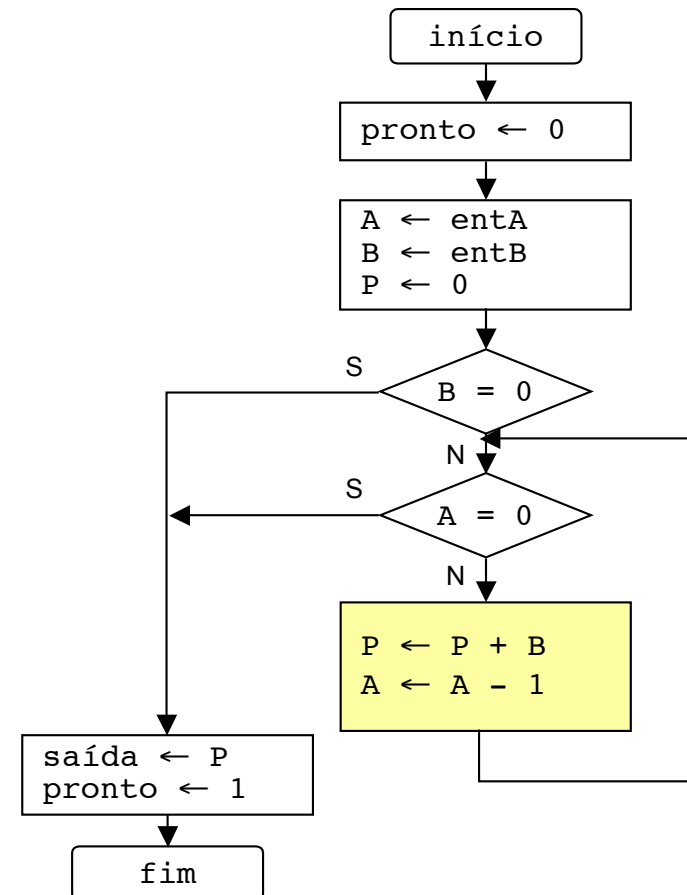
► Projeto do BO Visando Máximo Desempenho

Solução 2: Reestruturando o Algoritmo para máximo desempenho



```
início
pronto ← 0;
A ← entA;
B ← entB;
P ← 0;
Se B ≠ 0 então
Enquanto A ≠ 0 faça
  início
  P ← P + B;
  A ← A - 1;
  fim
saída ← P;
pronto ← 1;
fim
```

- Tentaremos realizar mais de uma operação por ciclo de relógio

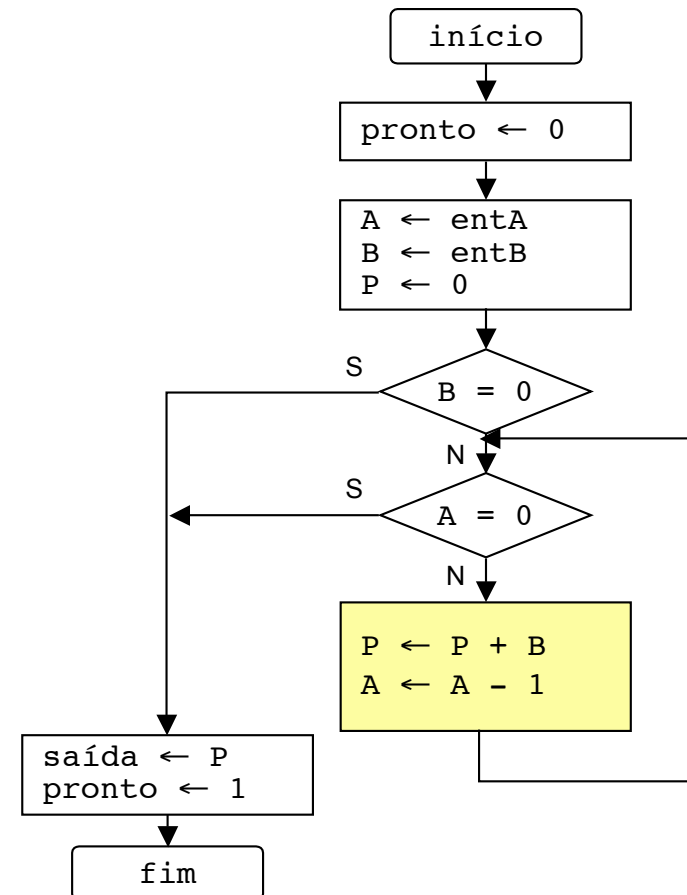


Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO Visando Máximo Desempenho

Solução 2: Unidades Funcionais (UFs) Necessárias

- As operações “+” e “-” serão realizadas no mesmo ciclo de relógio (em um único passo)
- Logo, necessitaremos de **um somador e um subtrator**



Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO Visando Máximo Desempenho

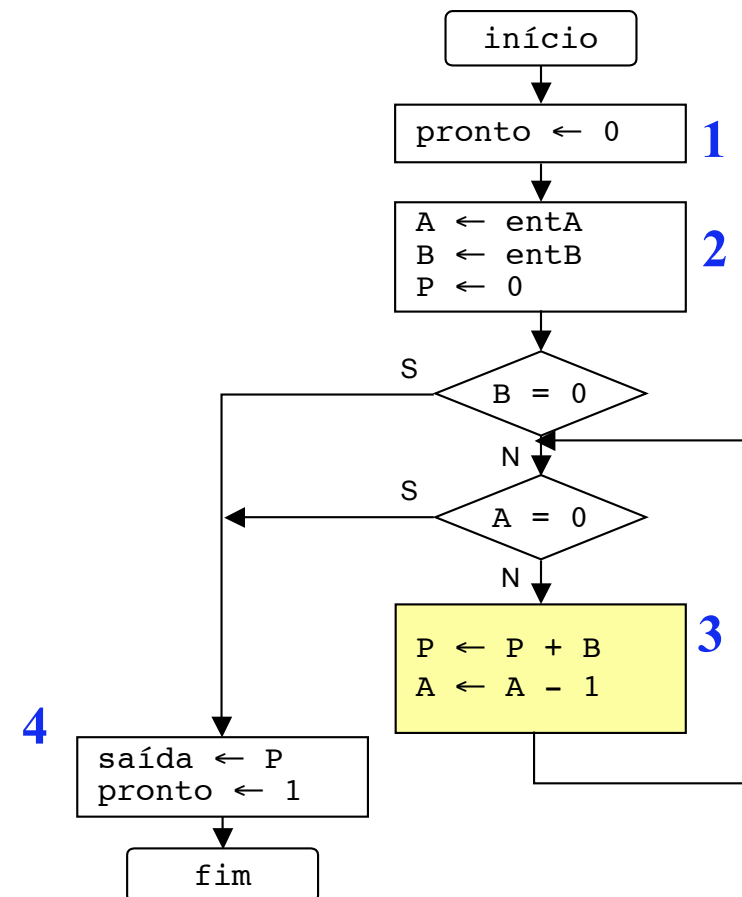
Solução 2: Registradores

Análise do tempo de vida das variáveis:

	1	2	3	4
A			X	
B			X	
P			X	X

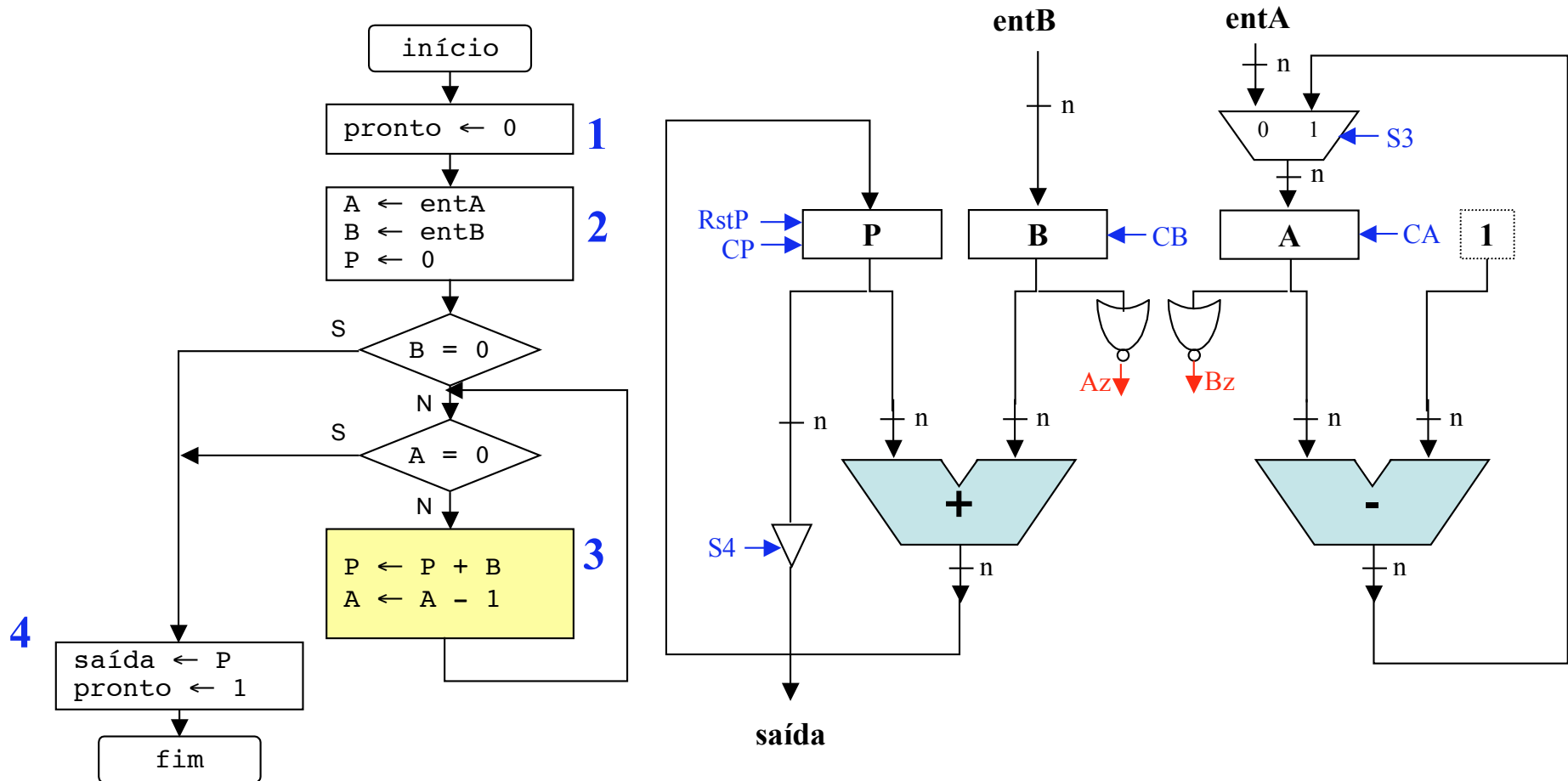
↑
as 3 variáveis são escritas no final do passo 2

Ainda são necessários **3 registradores (A, B e P)**



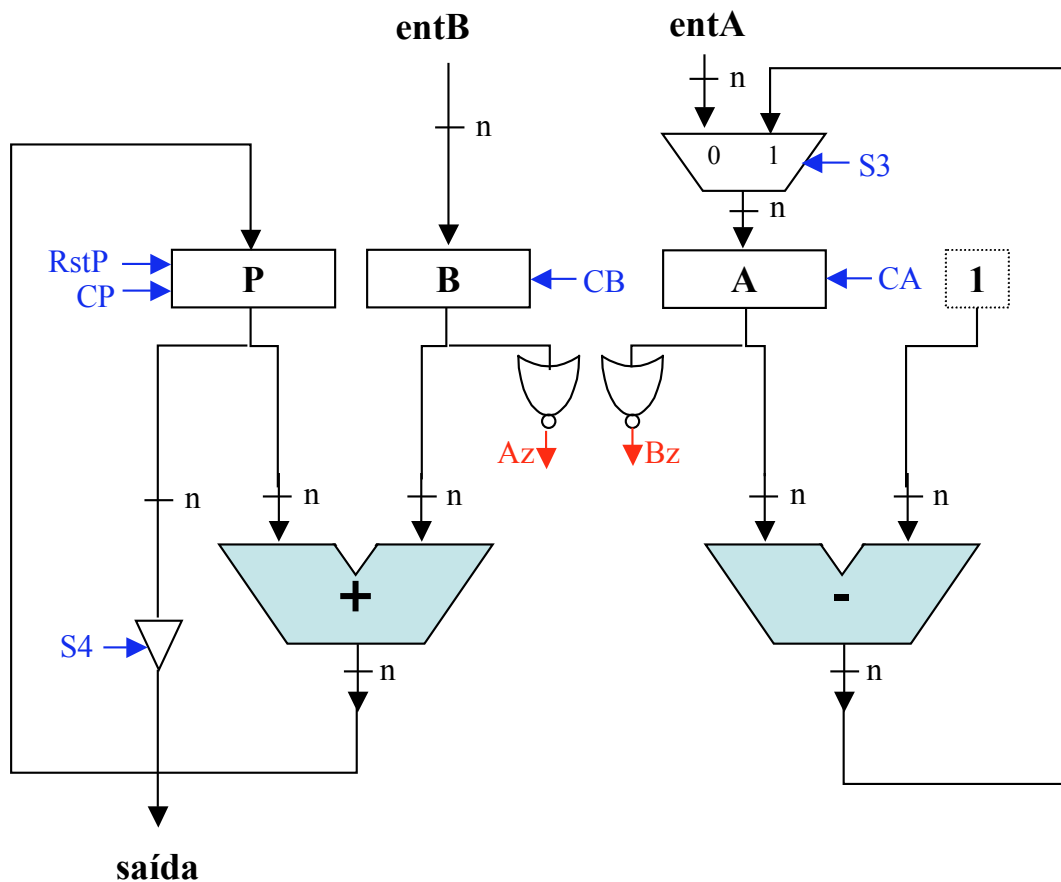
Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO Visando Máximo Desempenho



Projeto de Sistemas Digitais no Nível RT

▶ Estimativa do Custo do BO da Solução 2



Custo do BO 2	Custo
1 Somador	24n
1 Subtrator	26n
1 Muxes 2:1	4n
2 Registradores com carga paralela controlada	2x22n=44n
1 Registrador com carga paralela controlada e reset assíncrono	24n
1 conjunto de <i>buffers tri-state</i> não inversores	6n
Total	128n

Estimativa de custo para o BC:

- Número de estados: 4 ou 5
- Número de sinais de controle = 6

Projeto de Sistemas Digitais no Nível RT

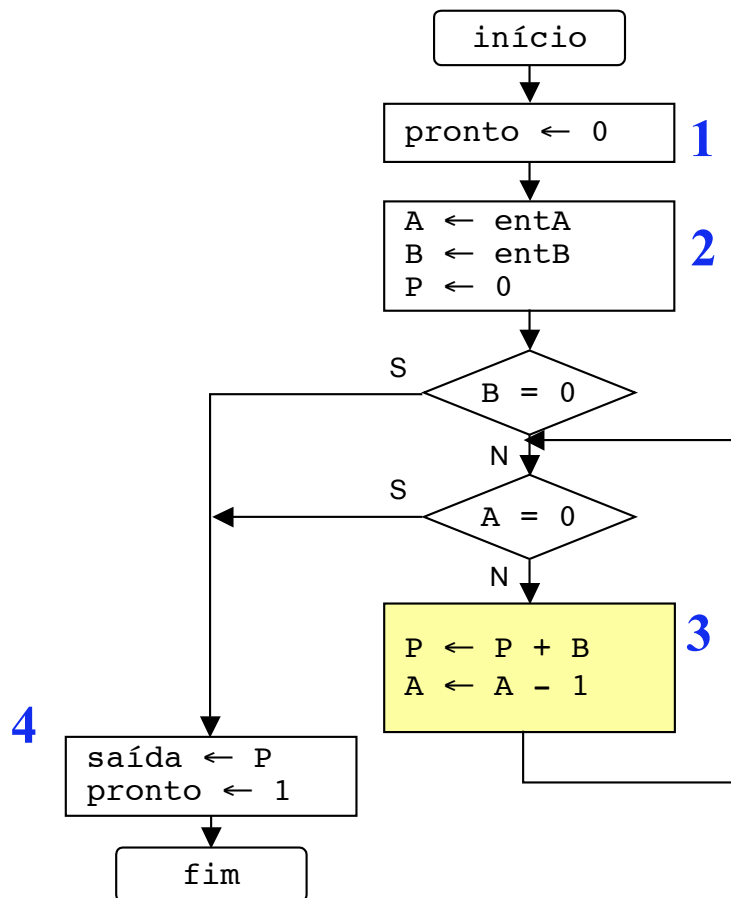
▶ Estimativa do Desempenho do BO da Solução 2

Se $n = 4$ bits:

- Maior inteiro sem sinal: 15 ($\Rightarrow 1111$)
- Pior caso: $A=15, B \neq 0$
- Seqüência de execução: 1, 2, $15 \times (3), 4$
= 18 passos (**18 ciclos de relógio**)
- BO 1 = 33 ciclos

Generalizando para n bits:

- Maior inteiro sem sinal: $2^n - 1$
- Pior caso: $A = 2^n - 1, B \neq 0$
- Seqüência de execução: 1, 2, $(2^n - 1) \times (3), 4$
= $(2^n - 1) + 3$ passos (\approx **2^n ciclos de relógio**)
- BO 1 = $\sim 2^{n+1}$ ciclos de relógio



Projeto de Sistemas Digitais no Nível RT

► Comparação Solução 1 x Solução 2

Quesito	BO 1	BO 2
Característica	Custo mínimo	Máximo desempenho
Custo (nº de transistores)	116n	128n
Tempo de Execução (nº de ciclos de relógio)	2^{n+1}	2^n
Impacto no BC nº de estados nº de sinais de controle	5 (6) 9 (5)	4 (5) 6 (4)

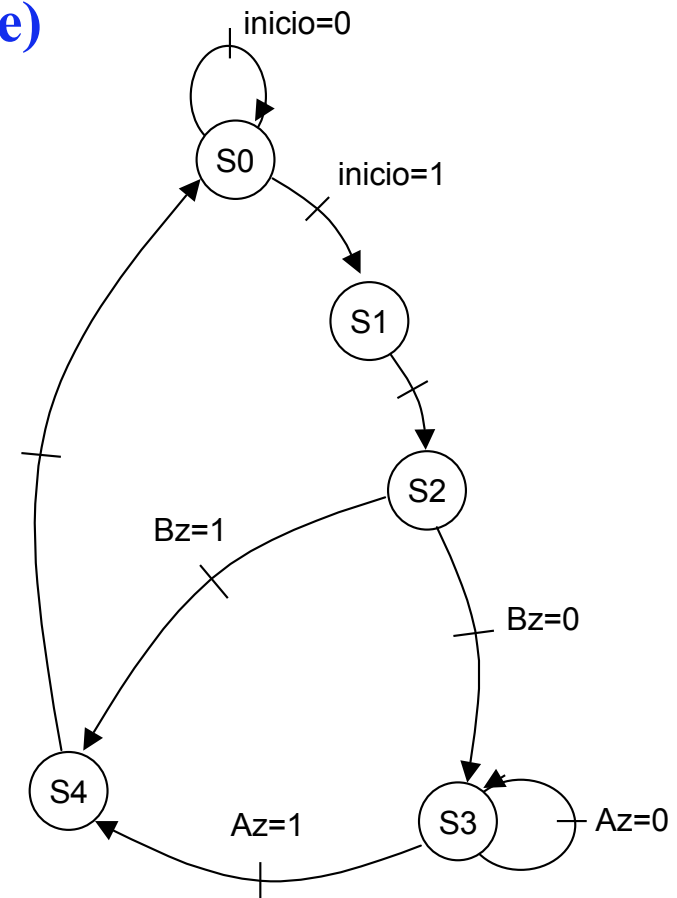
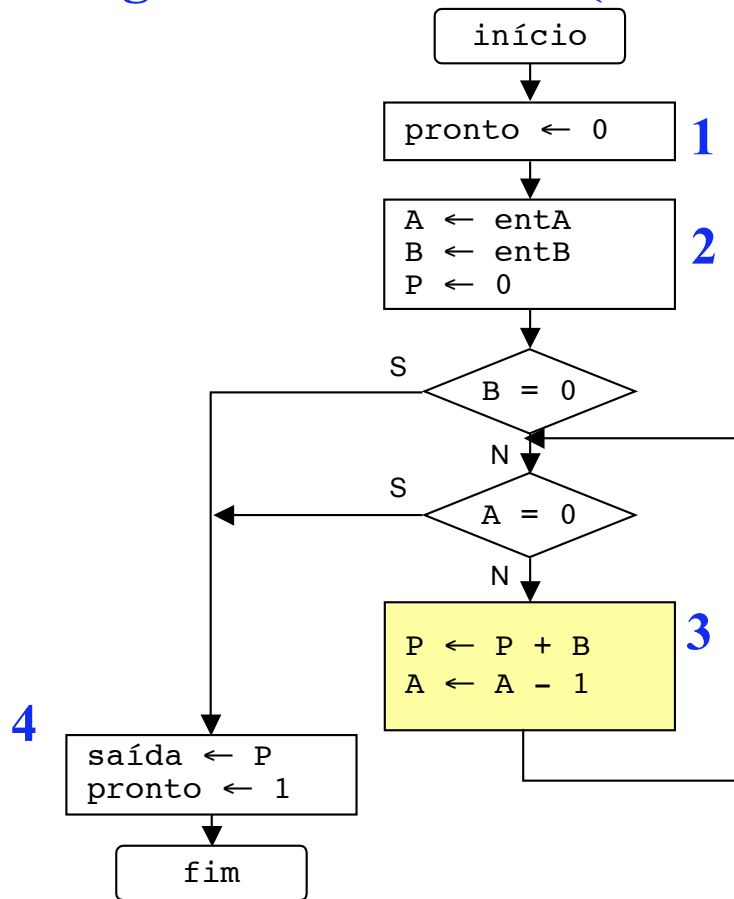
A exploração do paralelismo inerente ao algoritmo resultou em:

- Redução do número de passos de execução (redução do número de estados). No caso estudado, a aceleração foi de 2x.
- Maior custo do BO. No caso estudado, +10%.
- Menor número de sinais de controle necessários (indício de redução do custo do BC)

Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BC para a Solução 2

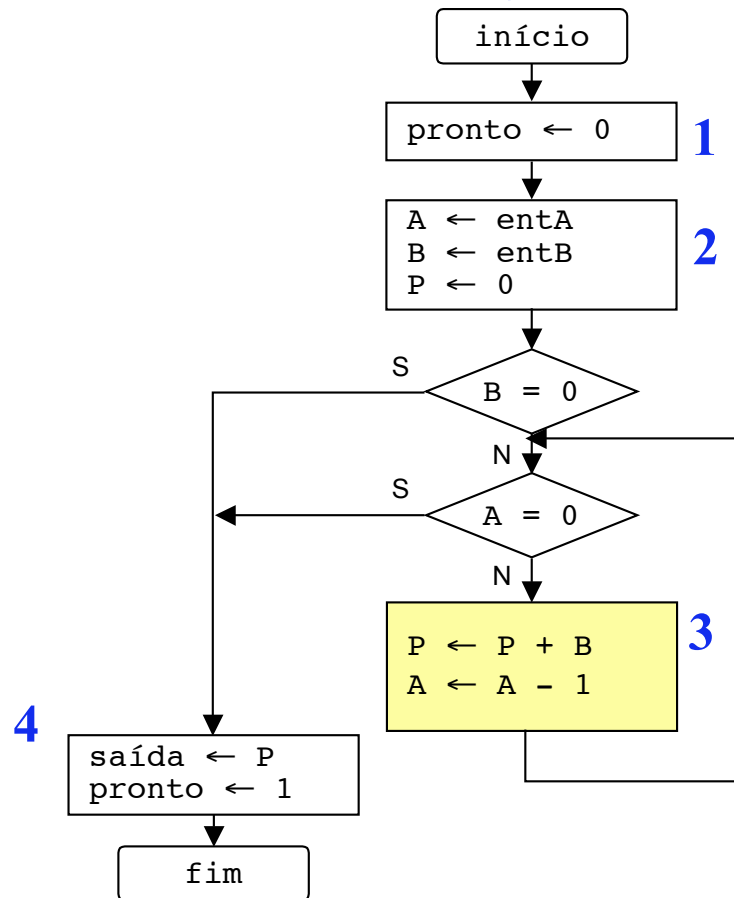
Diagrama de Estados (Assumindo Moore)



Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BC para a Solução 2

Tabela de Transição de Estados (Assumindo Moore)

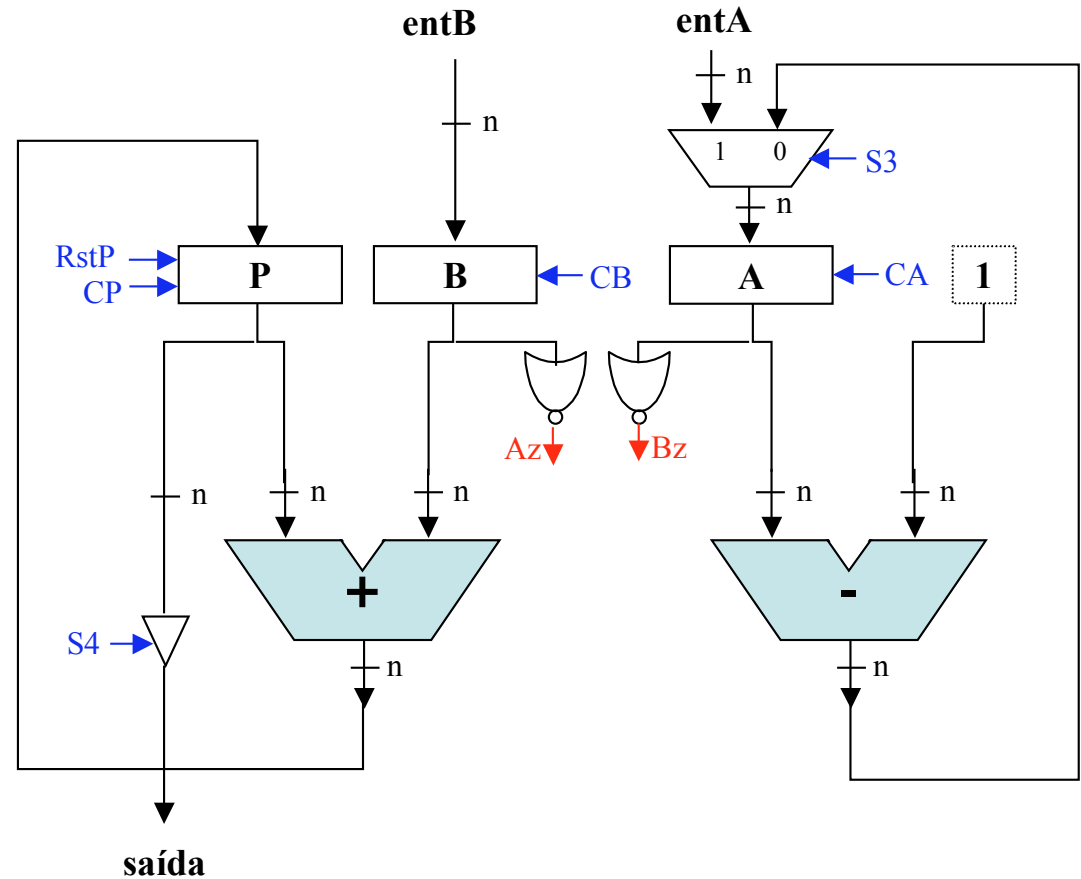
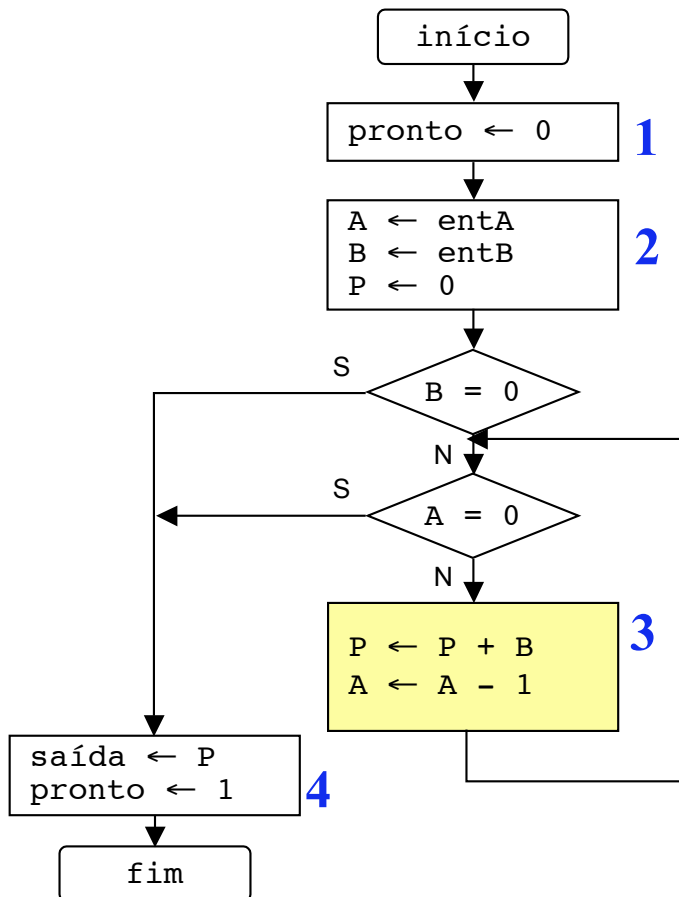


Estado atual	Entradas			Próx. Estado
	início	BZ	AZ	
S0	0	-	-	S0
	1	-	-	S1
S1	-	-	-	S2
S2	-	0	-	S3
	-	1	-	S4
S3	-	-	0	S3
	-	-	1	S4

Projeto de Sistemas Digitais no Nível RT

► Projeto do BC para a Solução 2

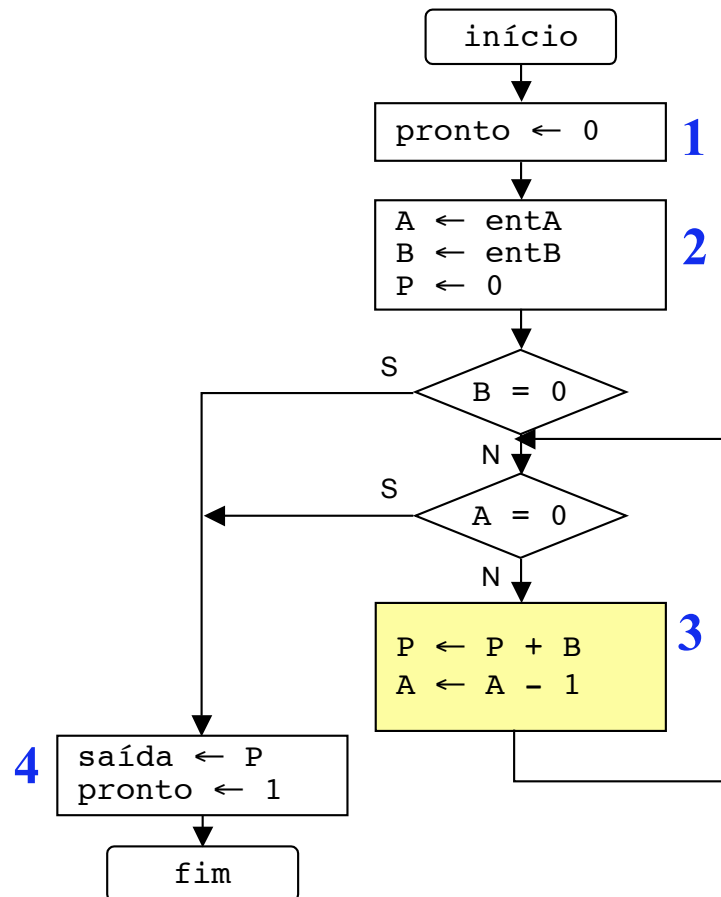
Tabela de Saídas (Assumindo Moore)



Projeto de Sistemas Digitais no Nível RT

► Projeto do BC para a Solução 2

Tabela de Saídas (Assumindo Moore)



Estado	Reg. P		Reg. A			Saída	
	RstP	CP	S3	CA	CB	S4	pronto
S0	0	0	-	0	0	0	-
S1	0	0	-	0	0	0	0
S2	1	0	1	1	1	0	0
S3	0	1	0	1	0	0	0
S4	0	0	-	0	0	1	1

1 sinal

1 sinal

4 sinais

RstP = CB = S3
S4 = pronto

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

O Algoritmo utilizado nas Soluções 1 e 2 não é eficiente

- Considerando a solução 2 e $n=4$ bits:
 - $A=1$ e $B=15$ (1×15) executa em 4 passos
 - $A=15$ e $B=1$ (15×1) executa em **18** passos
- **Solução:** projetar outro algoritmo, tentando explorar características inerentes ao problema a ser resolvido...
- **Exigência:** necessário conhecer detalhadamente o problema a ser resolvido

```
início
  pronto ← 0;
  A ← entA;
  B ← entB;
  P ← 0;
  Se B ≠ 0 então
  Enquanto A ≠ 0 faça
    início
      P ← P + B;
      A ← A - 1;
    fim
  saída ← P;
  pronto ← 1;
fim
```

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

Multiplicação de Inteiros (Binários) Sem Sinal

Exemplos Numéricos:

Com Decimais

$$\begin{array}{r} \\ x \\ \hline + \\ \\ \hline \\ \\ \hline \end{array} \begin{array}{l} \text{multiplicando} \\ \text{multiplicador} \\ \left. \begin{array}{l} \text{produtos} \\ \text{parciais} \end{array} \right\} \\ \text{resultado} \end{array}$$

Com Binários

$$\begin{array}{r} \\ x \\ \hline + \\ \\ \hline \\ \\ \hline \end{array} \begin{array}{l} \text{multiplicando} \\ \text{multiplicador} \\ \left. \begin{array}{l} \text{produtos} \\ \text{parciais} \end{array} \right\} \\ \text{resultado} \end{array}$$

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

Multiplicação de Inteiros Binários Sem Sinal: o algoritmo de somas e deslocamentos

Explorando as características do problema:

- Gerar n produtos parciais
- Somar n produtos parciais
- n = número de bits do multiplicando
(logo, tempo de execução independe dos dados, exceto quando operando =0)

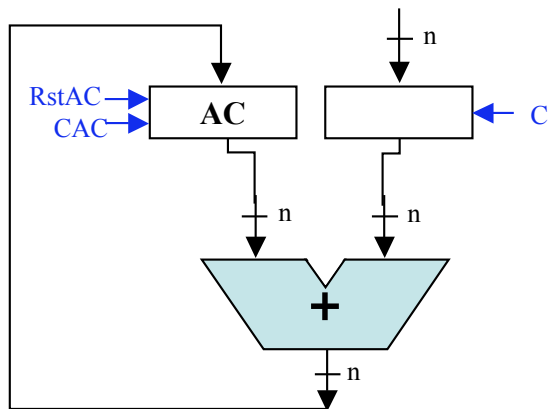
		1 0 0 1	multiplicando
	x	1 0 1 1	multiplicador
		<hr/>	
		1 0 0 1	} produtos parciais
	+	1 0 0 1 -	
		0 0 0 0 - -	
		1 0 0 1 - - -	
		<hr/>	
		1 1 0 0 0 1 1	resultado

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

Multiplicação de Inteiros Binários Sem Sinal: o algoritmo de somas e deslocamentos

- Problema: somador para 4 número de n bits é demasiado caro
- Solução: realizar n passos de soma (usar acumulador para somar produtos parciais devidamente deslocados)



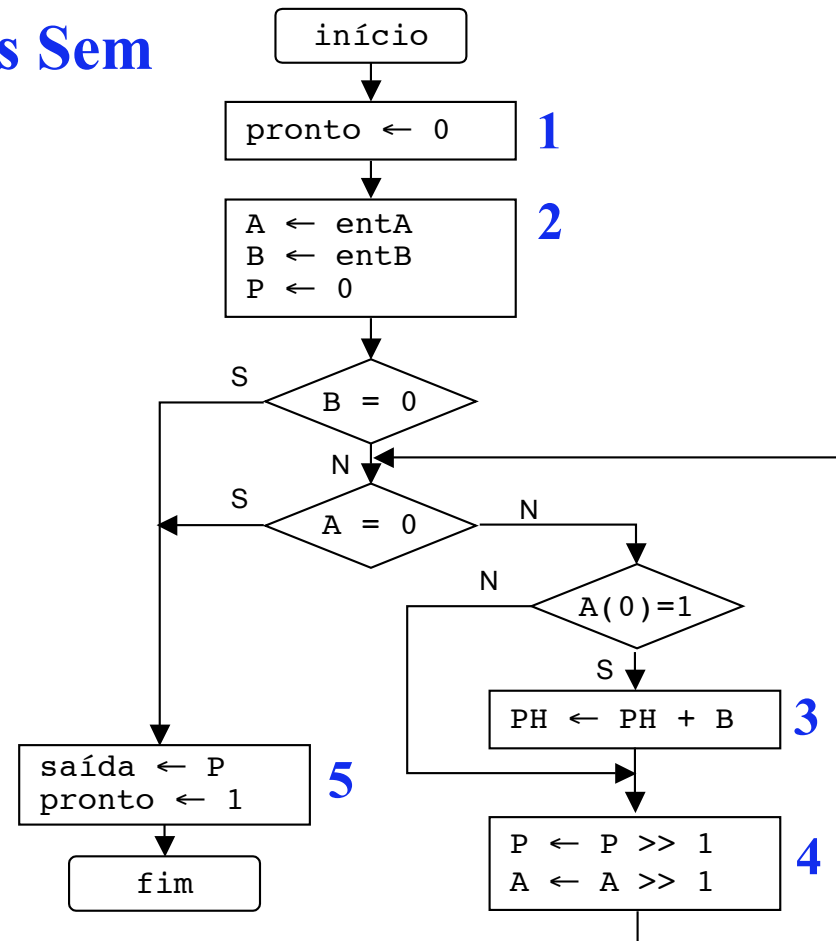
x	1 0 0 1	multiplicando
	1 0 1 1	multiplicador
+	1 0 0 1	P1
	0 0 0 0	AC
+	1 0 0 1	AC
	1 0 0 1 -	P2
+	1 1 0 1 1	AC
	0 0 0 0 - -	P3
+	0 1 1 0 1 1	AC
	1 0 0 1 - - -	P4
	1 1 0 0 0 1 1	AC=resultado

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

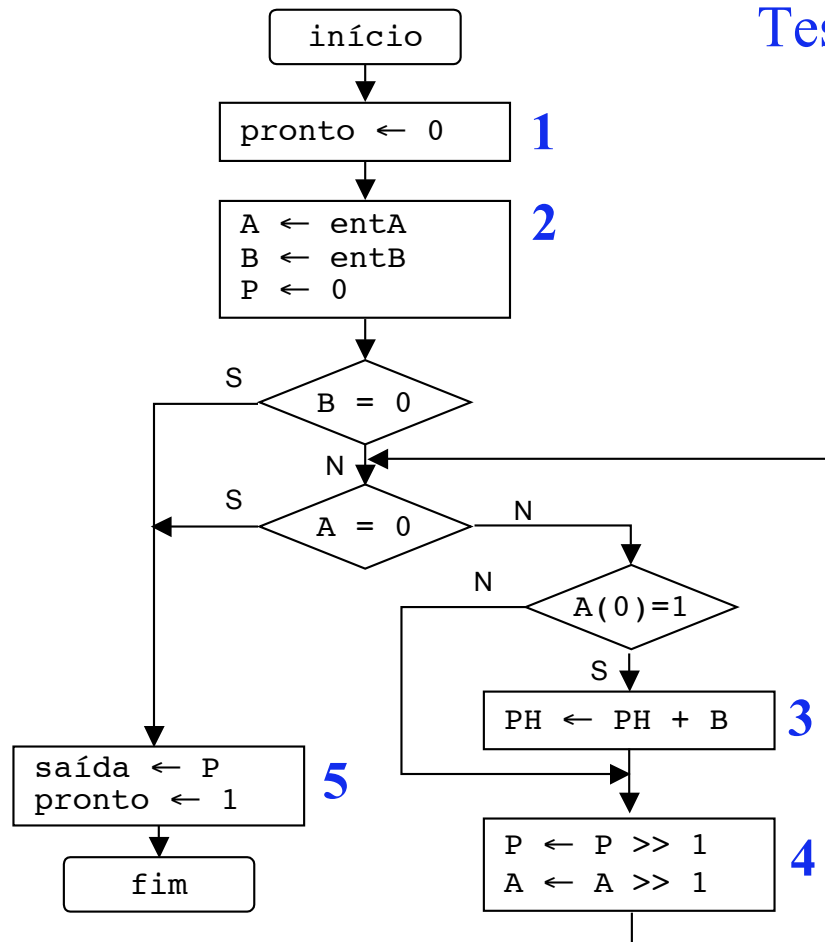
Multiplicação de Inteiros Binários Sem Sinal: o algoritmo de somas e deslocamentos

- A variável **P** irá usar um registrador com $2n$ bits, o qual será dividido em parte alta (**PH**) e parte baixa (**PL**), cada uma com n bits (não haverá problema de *overflow*)
- $A(0)$ é o bit menos significativo de **A**
- $P \gg 1$ significa deslocar o conteúdo de **P** um bit para a direita (possivelmente, injetando um “0” pela esquerda)



Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica

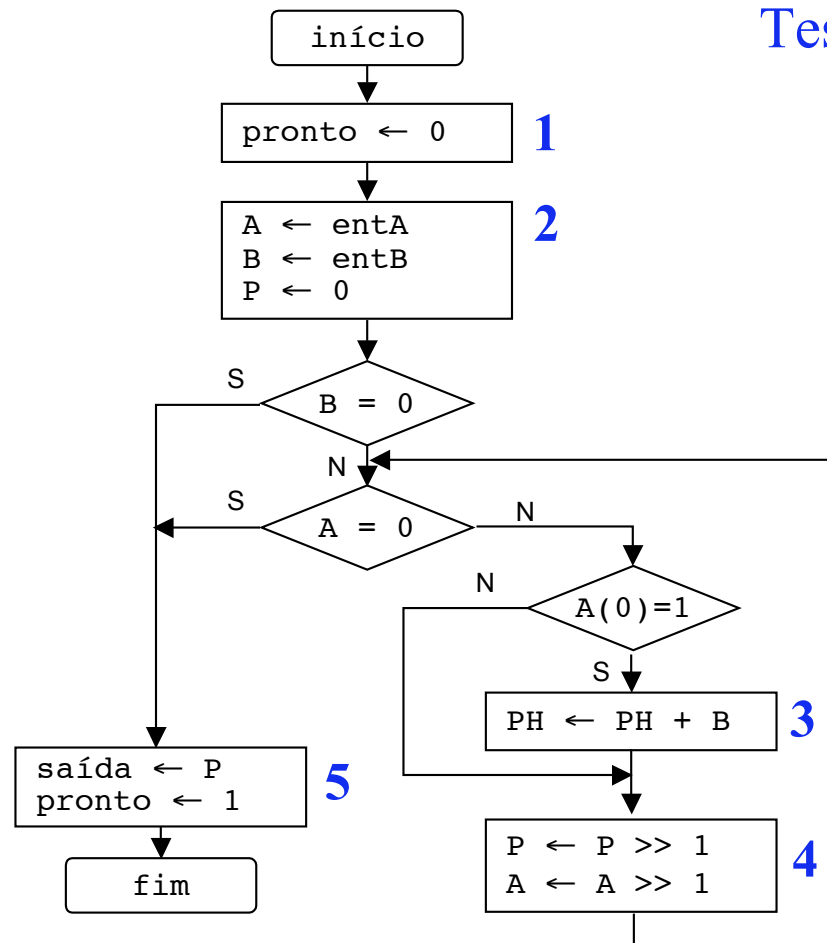


Teste de Mesa do Algoritmo de Multiplicação por Somas e Deslocamentos

	A	B	PH	PL
2	1011	1001	0000	0000
3.1				
4.1				
3.2				
4.2				
4.3				
3.4				
4.4				

Projeto de Sistemas Digitais no Nível RT

► Exploração Algorítmica



Teste de Mesa do Algoritmo de Multiplicação por Somas e Deslocamentos

	A	B	PH	PL
2	1011	1001	0000	0000
3.1	-	-	1001	0000
4.1	0101	-	0100	1000
3.2	-	-	1101	1000
4.2	0010	-	0110	1100
4.3	0001	-	0011	0110
3.4	-	-	1100	0110
4.4	0000	-	0110	0011

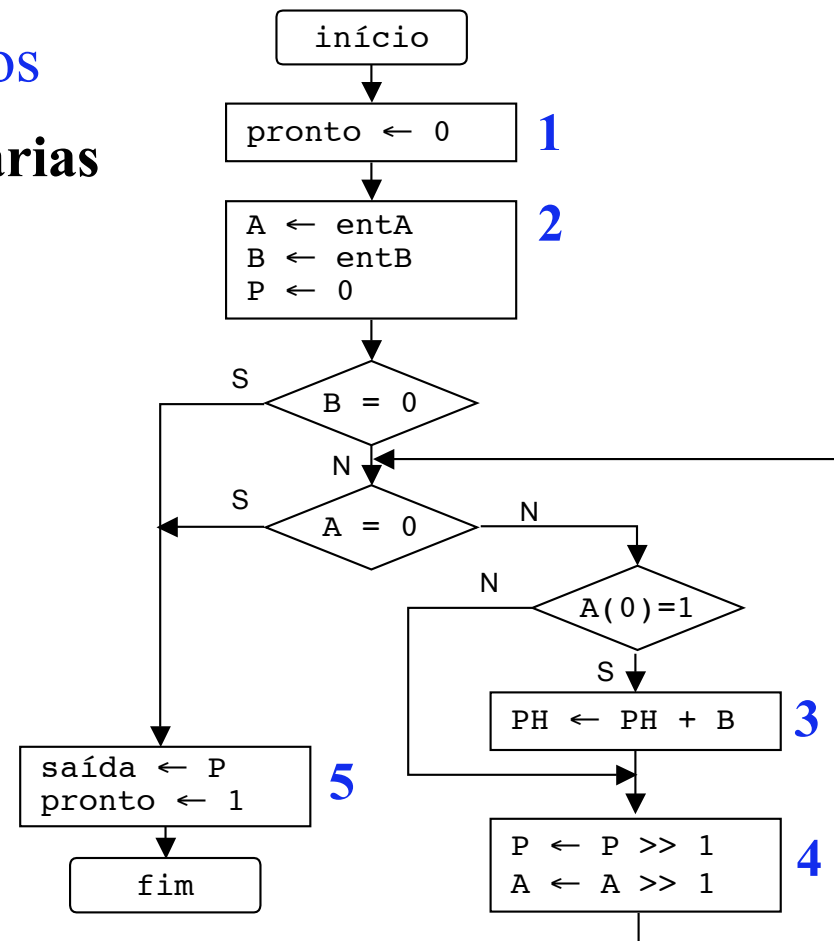
Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO 3

Solução 3: Somas e Deslocamentos

Unidades Funcionais (UFs) Necessárias

- Para os deslocamentos, iremos usar registradores de deslocamento
- Logo, necessitaremos de **um somador** (apenas)



Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO 3

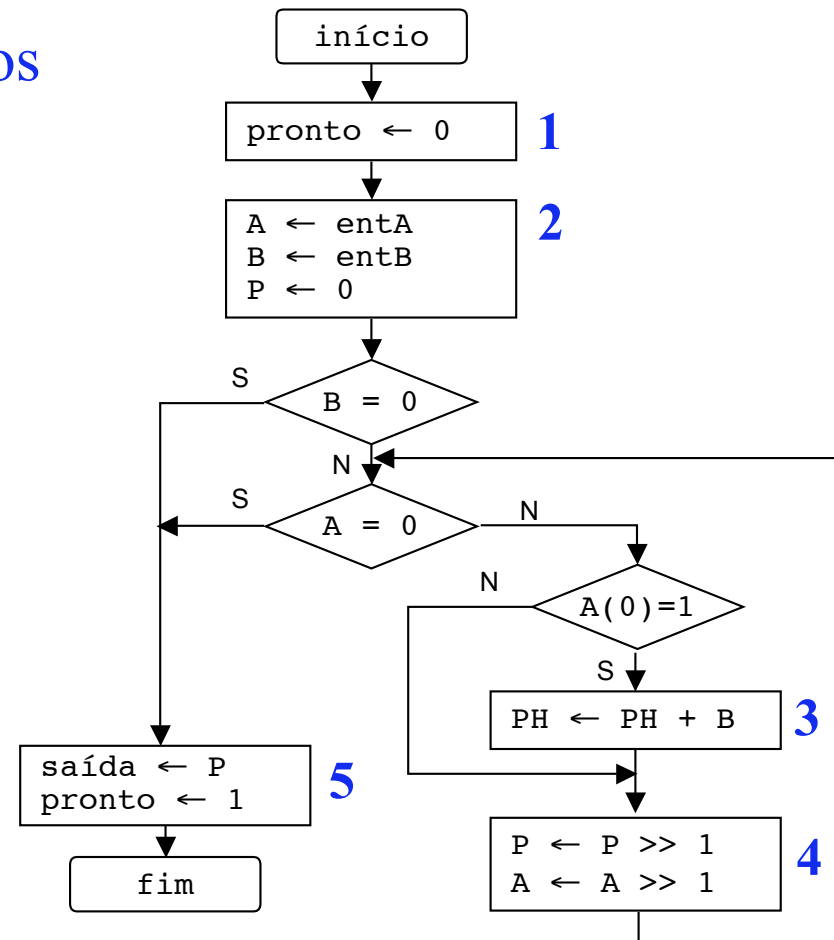
Solução 3: Somas e Deslocamentos

Análise do tempo de vida das variáveis:

	1	2	3	4	5
A			X	X	
B			X	X	
P			X	X	X

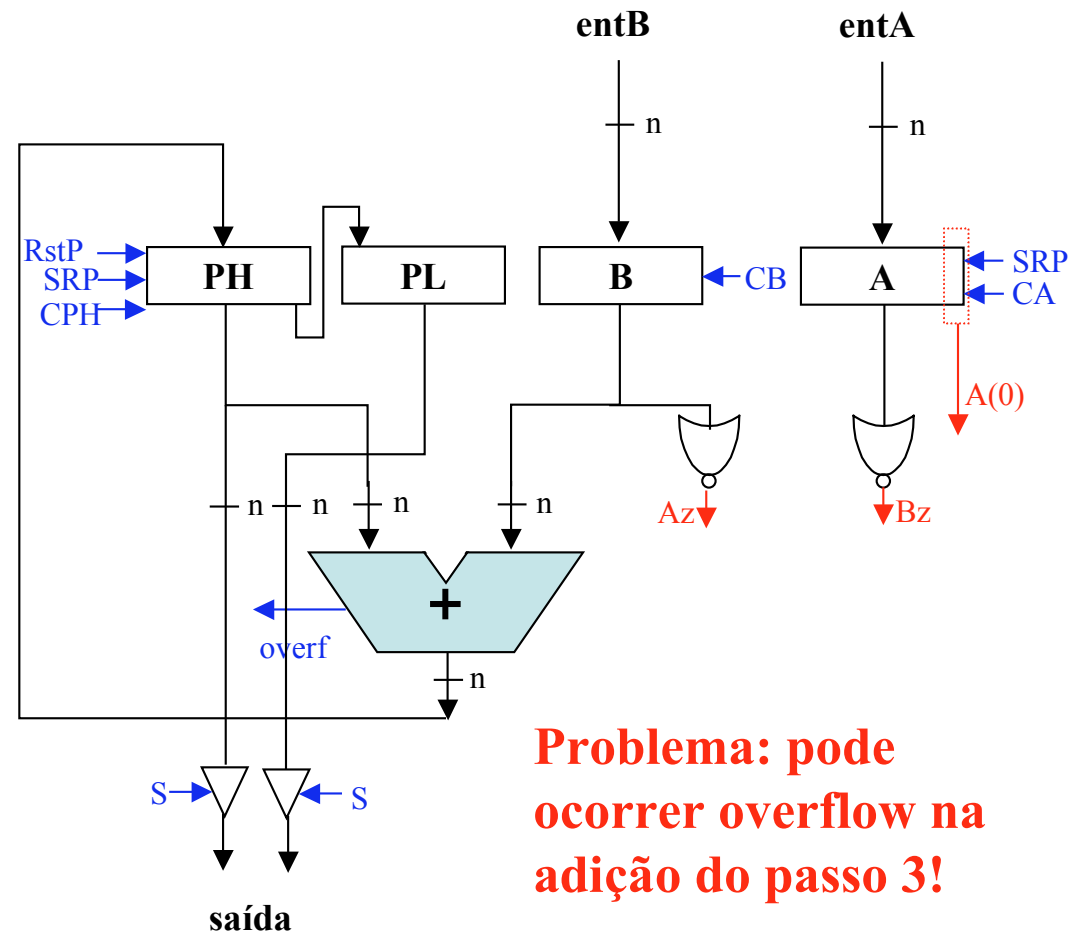
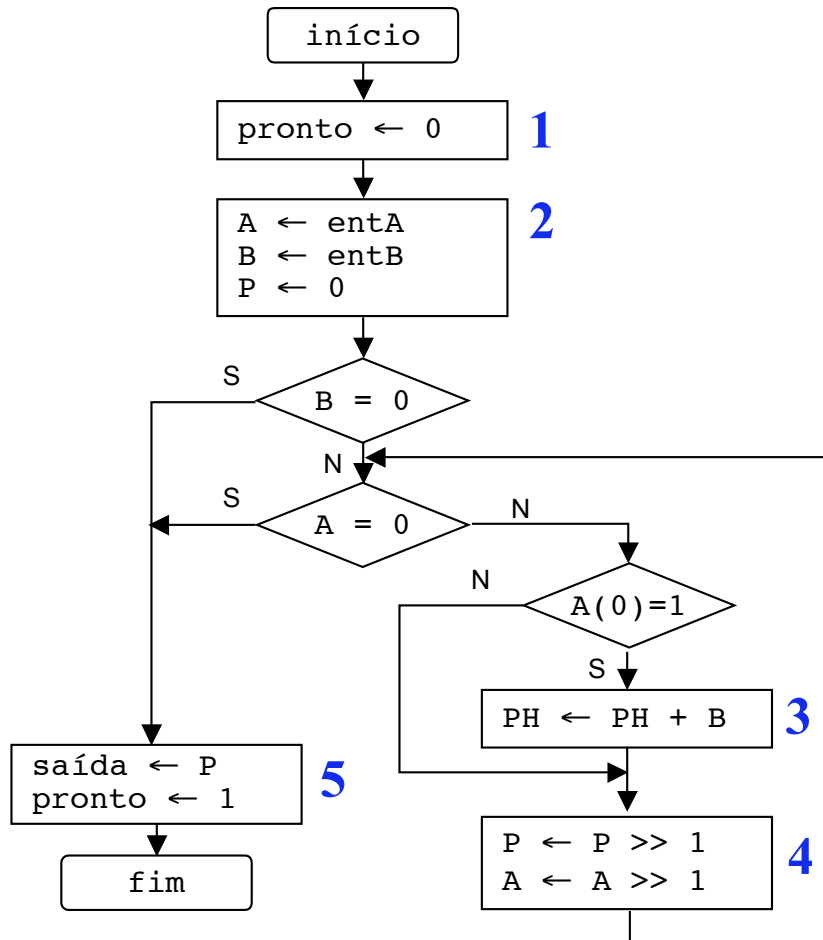
↑
as 3 variáveis são escritas no final do passo 2

Ainda são necessários **3 registradores (A, B e P)**



Projeto de Sistemas Digitais no Nível RT

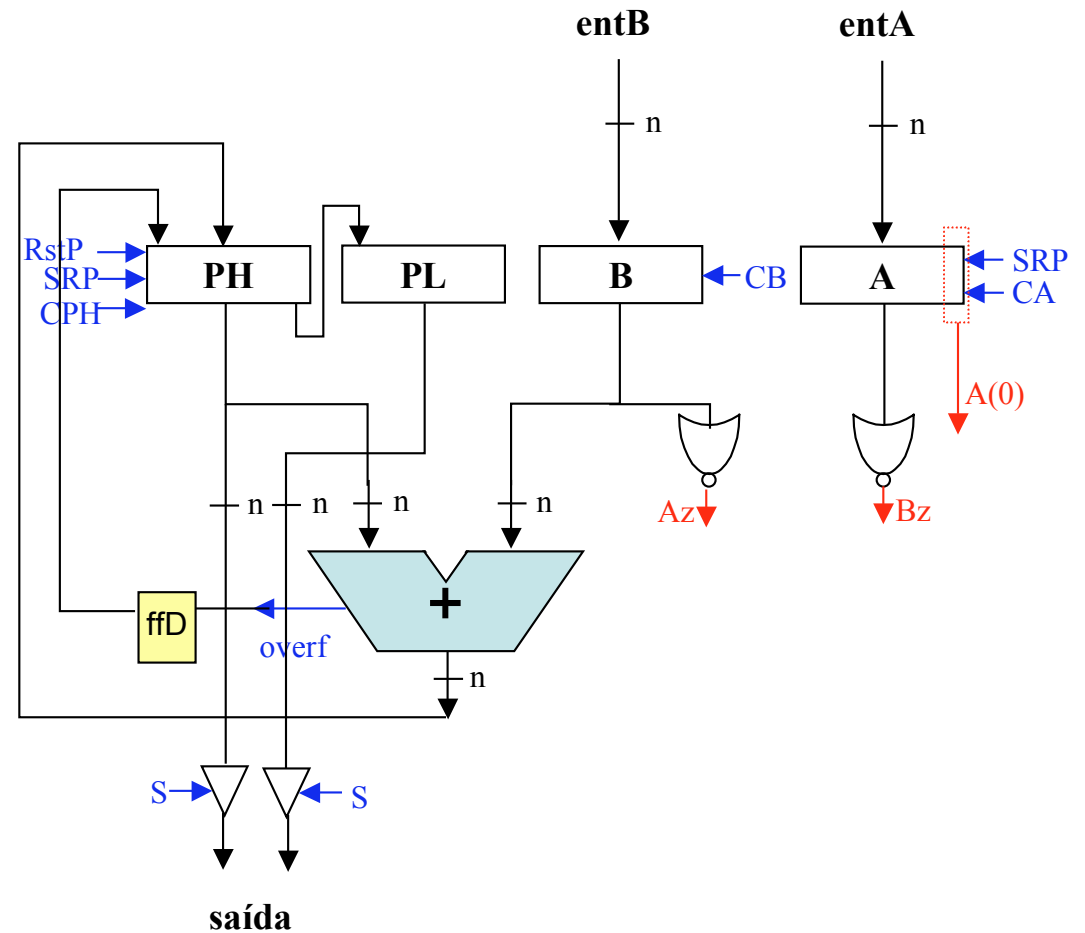
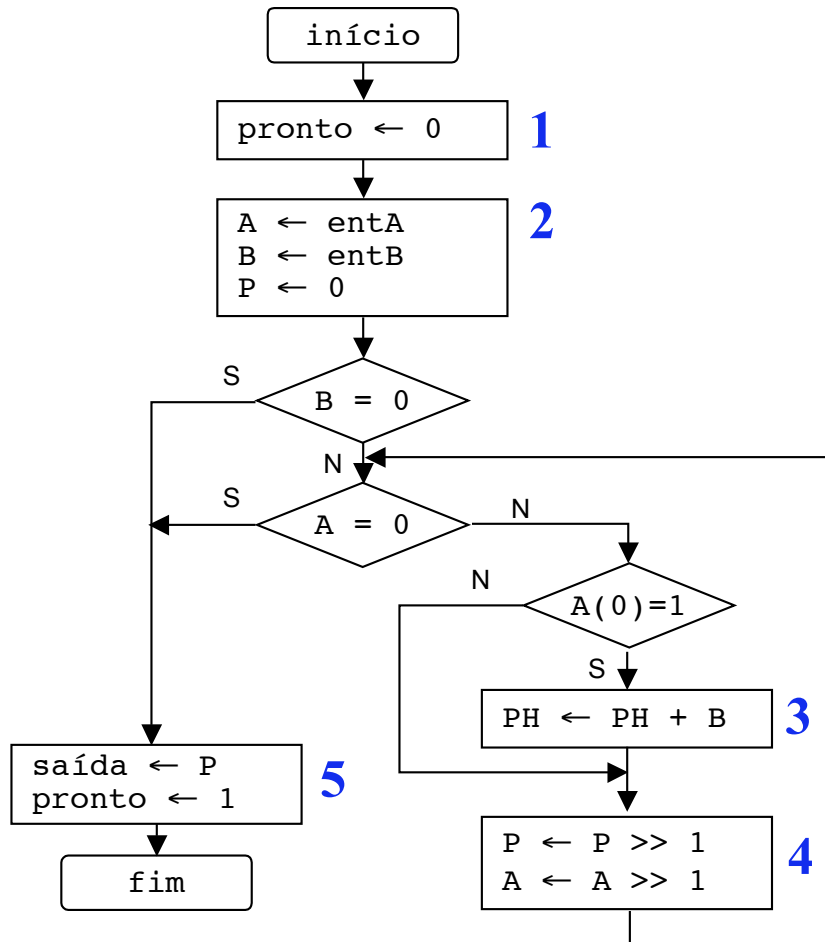
▶ Projeto do BO 3



Problema: pode ocorrer overflow na adição do passo 3!

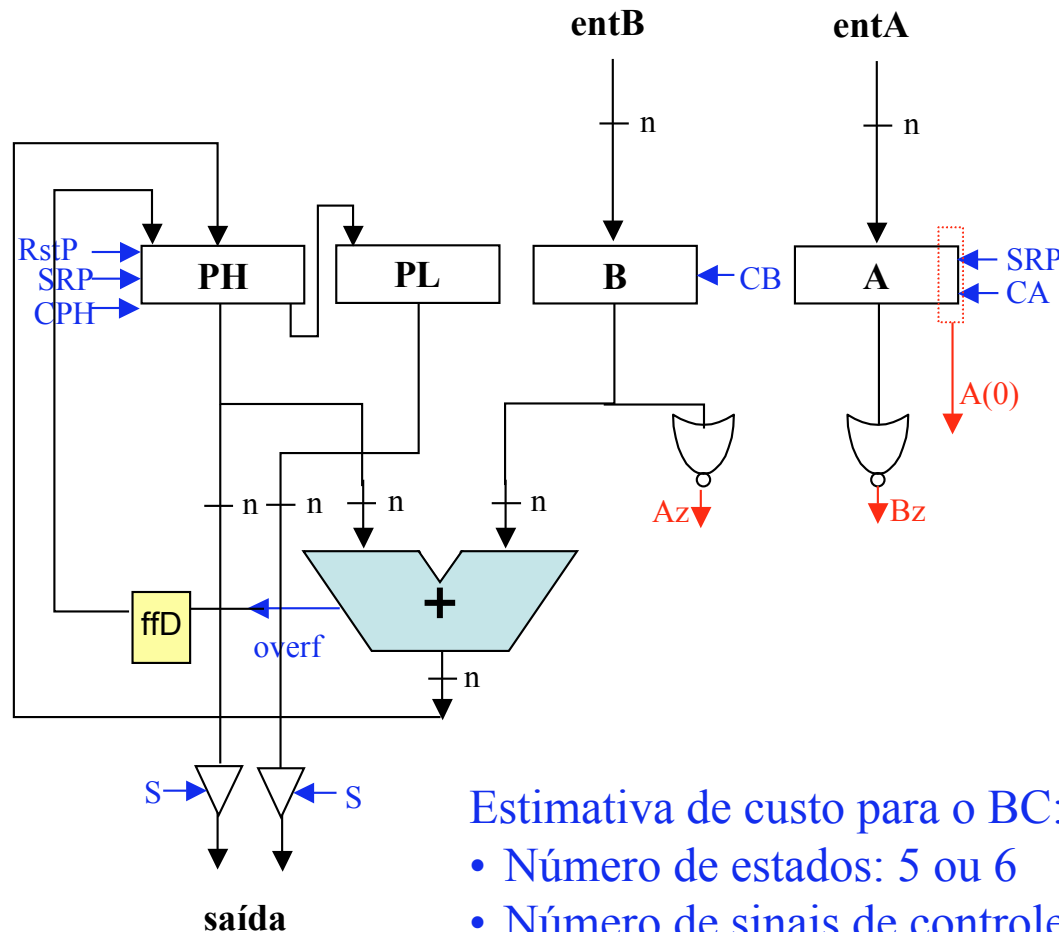
Projeto de Sistemas Digitais no Nível RT

▶ Projeto do BO 3



Projeto de Sistemas Digitais no Nível RT

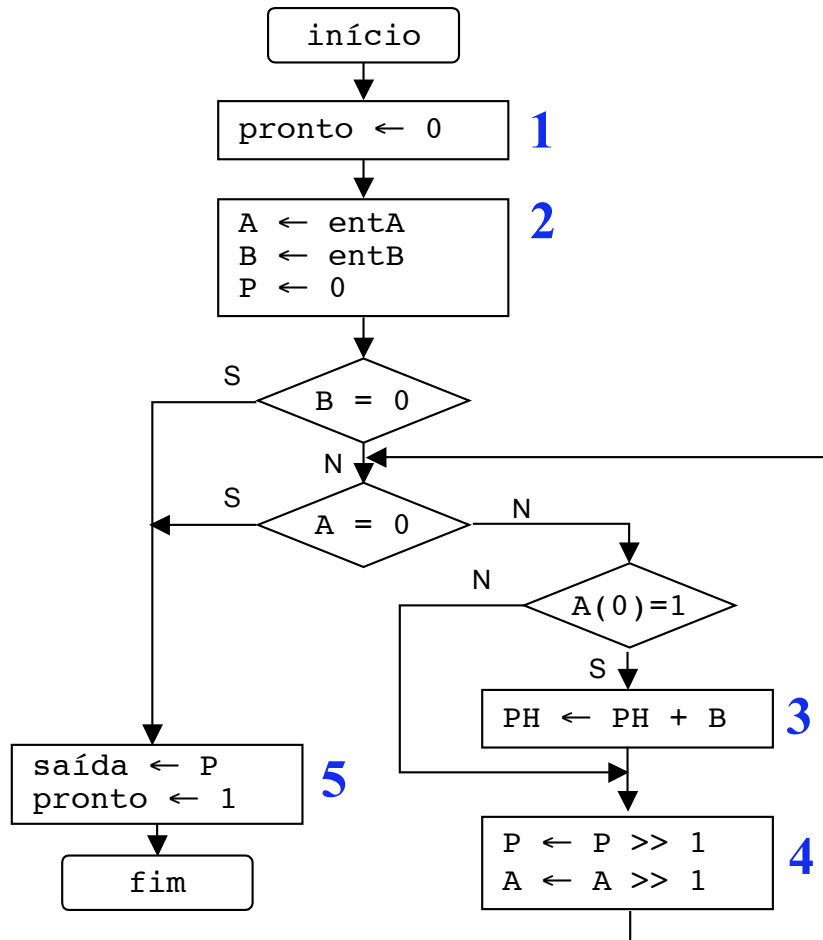
▶ Estimativa do Custo do BO da Solução 3



Custo do BO 3	Custo
1 Somador	24n
1 Registrador com carga paralela controlada (B)	22n
1 Registrador de deslocamento com carga paralela controlada (A)	24n
1 Registrador de deslocamento com carga paralela controlada e reset assíncrono	26n
1 Registrador de deslocamento	22n
2 conjunto de <i>buffers tri-state</i> não inversores	$2 \times 6n = 12n$
Total	120n

Projeto de Sistemas Digitais no Nível RT

▶ Estimativa do Desempenho do BO da Solução 3



Se $n = 4$ bits:

- Maior inteiro sem sinal: 15 ($\Rightarrow 1111$)
- Pior caso: $A \neq 15, B \neq 0$
- Seqüência de execução: 1, 2, $4 \times (3,4), 5 = 11$ passos (**11 ciclos de relógio**)
- BO 1 = 33 ciclos, BO 2 = 18 ciclos

Generalizando para n bits:

- Maior inteiro sem sinal: $2^n - 1$
- Pior caso: $A \neq 0, B \neq 0$
- Seqüência de execução: 1, 2, $n \times (3,4), 5 = 2n + 3$ passos ($\approx 2n$ ciclos de relógio)
- BO 1 = $\sim 2^{n+1}$ ciclos de relógio, BO 2 = $\sim 2^n$ ciclos de relógio

Projeto de Sistemas Digitais no Nível RT

► Multiplicação com Circuito Combinacional

O Multiplicador Matricial

- É uma implementação direta do esquema ao lado
- Cada bit dos produtos parciais é gerado por meio de um “E” lógico

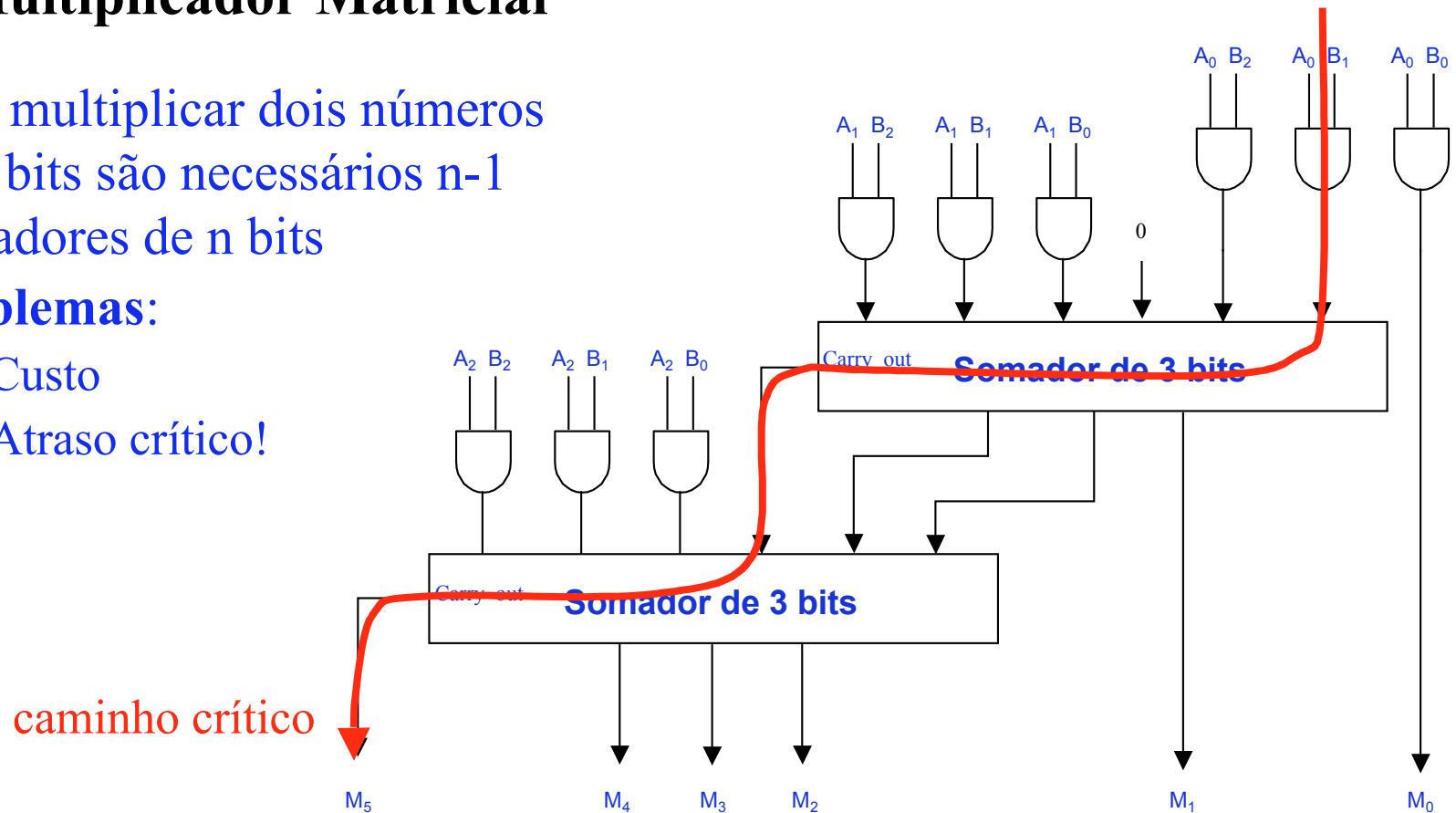
		1 0 0 1	multiplicando
x		1 0 1 1	multiplicador
<hr/>			
		1 0 0 1	} produtos parciais
+		1 0 0 1 -	
		0 0 0 0 - -	
		1 0 0 1 - - -	
<hr/>			
		1 1 0 0 0 1 1	resultado

Projeto de Sistemas Digitais no Nível RT

► Multiplicação com Circuito Combinacional

O Multiplicador Matricial

- Para multiplicar dois números de n bits são necessários $n-1$ somadores de n bits
- **Problemas:**
 - Custo
 - Atraso crítico!

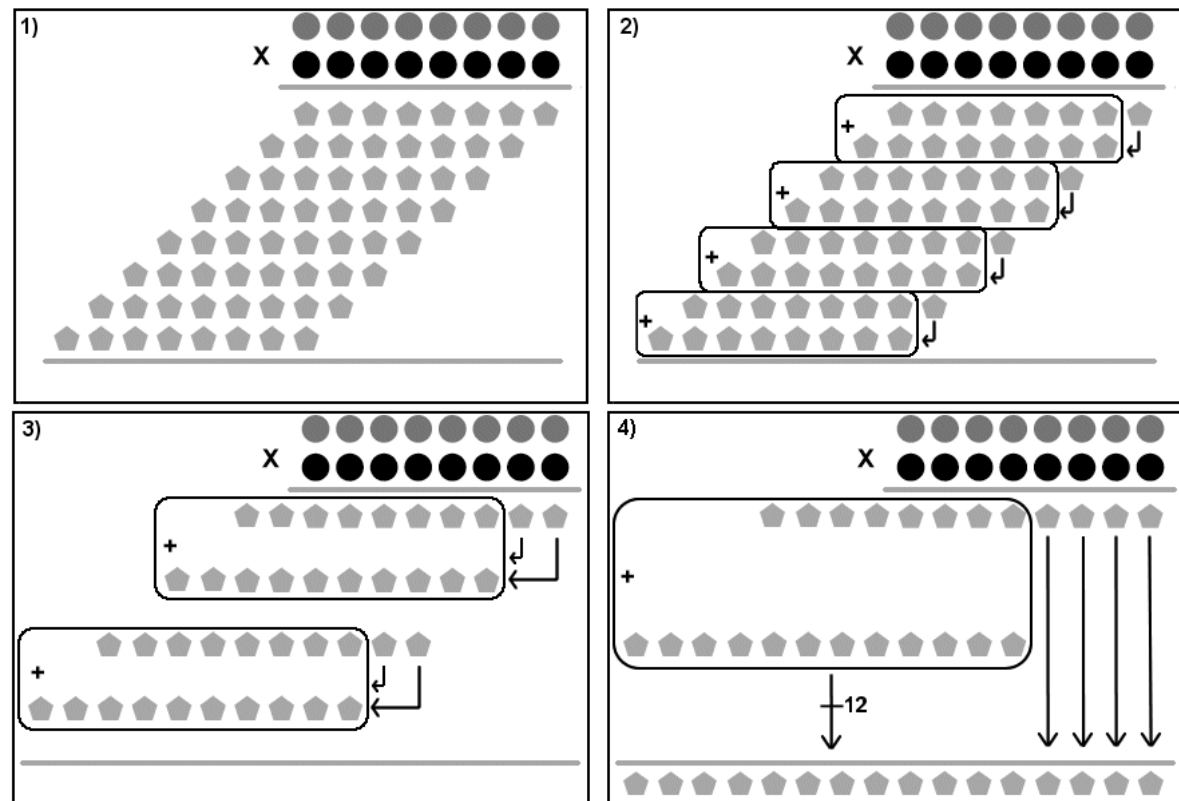


Projeto de Sistemas Digitais no Nível RT

► Multiplicação com Circuito Combinacional

O Multiplicador Matricial *Pipeline*

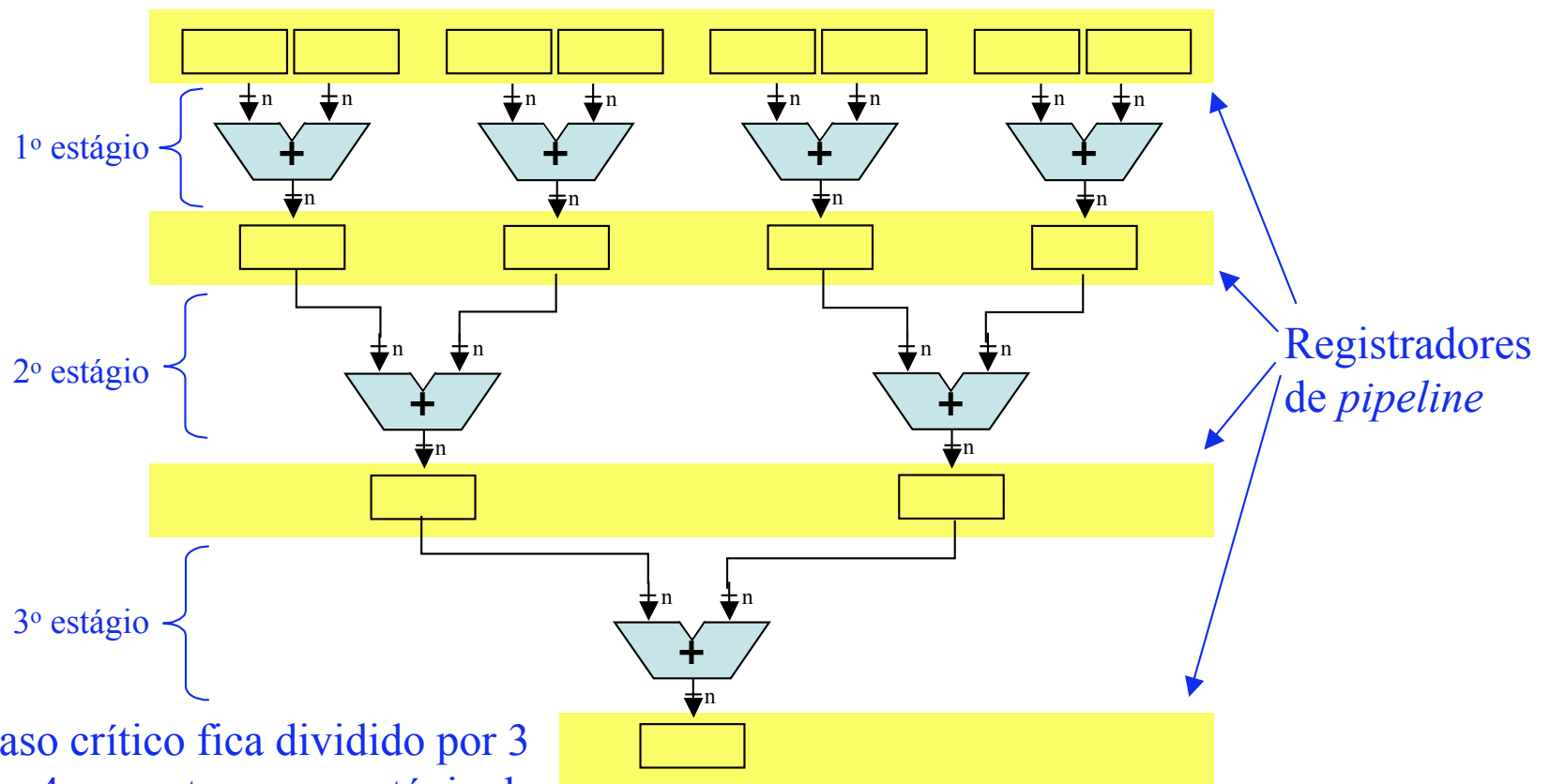
- **Passo 1:** todos os produtos parciais são gerados
- **Passo 2:** os produtos parciais são somados de dois em dois
- **Passo 3:** os resultados do passo anterior são somados de dois em dois
- ...



Projeto de Sistemas Digitais no Nível RT

► Multiplicação com Circuito Combinacional

O Multiplicador Matricial *Pipeline*



O Atraso crítico fica dividido por 3 (ou por 4 se contarmos o estágio de geração dos produtos parciais

Lógica Programável - semestre 2007/2

slide 11.30

Prof. José Luís Güntzel