



PCS 3115 – Sistemas Digitais I

Circuitos Combinatórios: Somadores e Subtratores

Suporte para EAD

Parte I:

Projeto de somadores como circuitos iterativos.

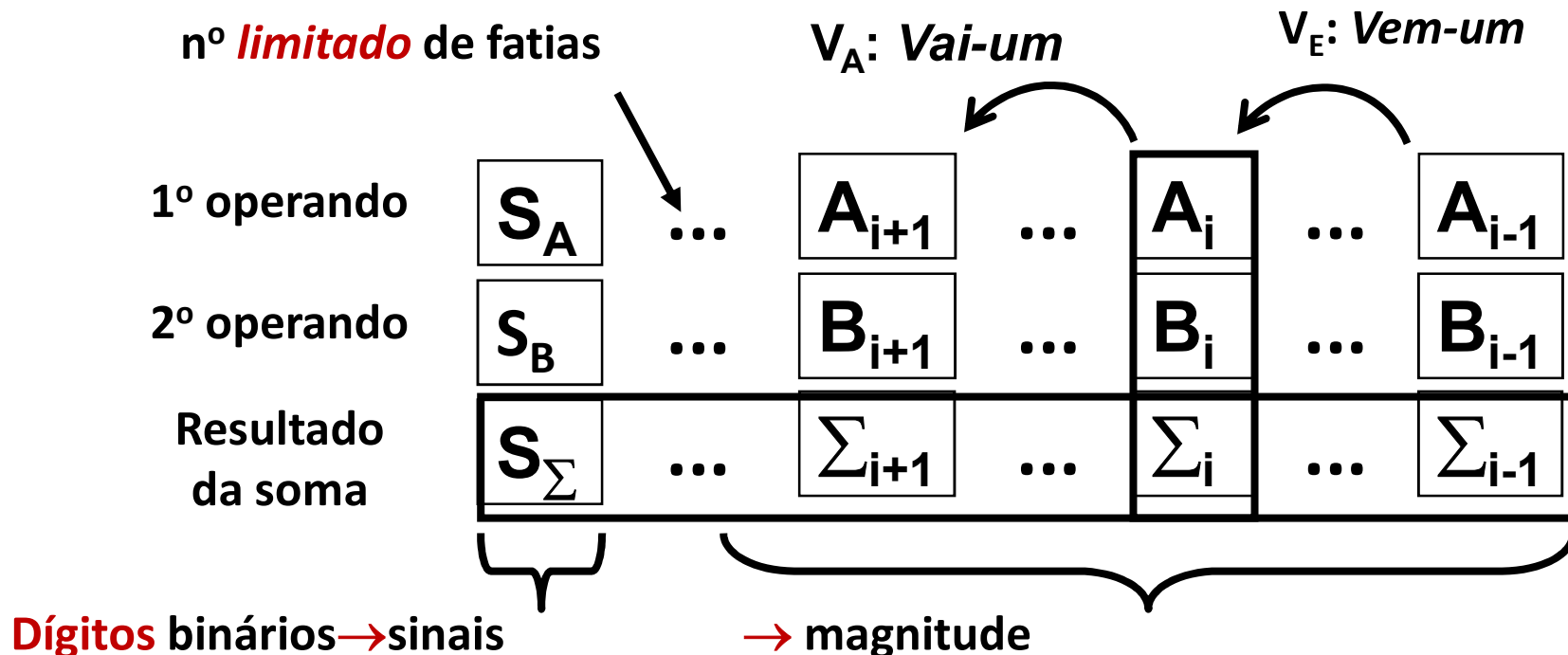
Aula: 14 – Data: 27/04 (S)

Prof. Dr. Marco Túlio Carvalho de Andrade

versão: 1.0 (Abril/2020)

Lembrando: Adição de Números Binários

- **Lembrando: Operações com números binários:** O resultado da adição é obtido calculando-se os resultados parciais, em *fatias* individuais, **iterativamente** ... Como fizemos no projeto de comparadores ... Depois realiza-se o cascadeamento dos módulos

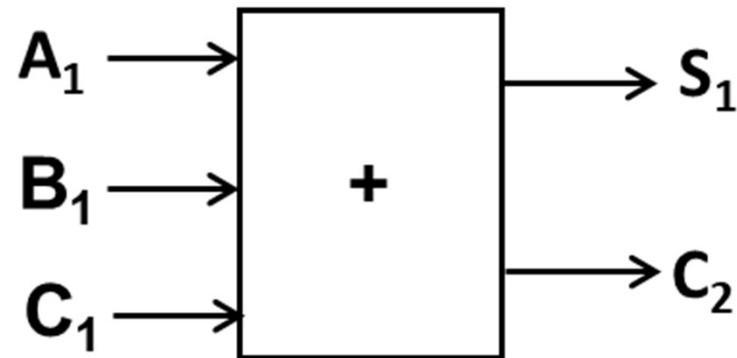


Somador

- Como construir um somador binário?
 - **Abordagem de Circuitos Iterativos:**
 - Projetar por fatias (faixas) e cascatear;
 - Definir o módulo básico.



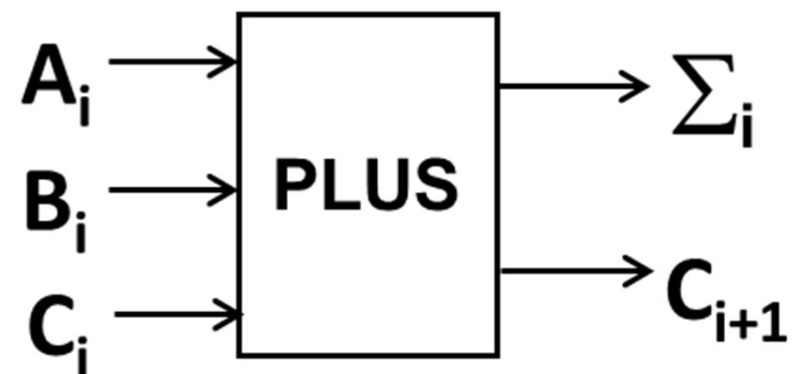
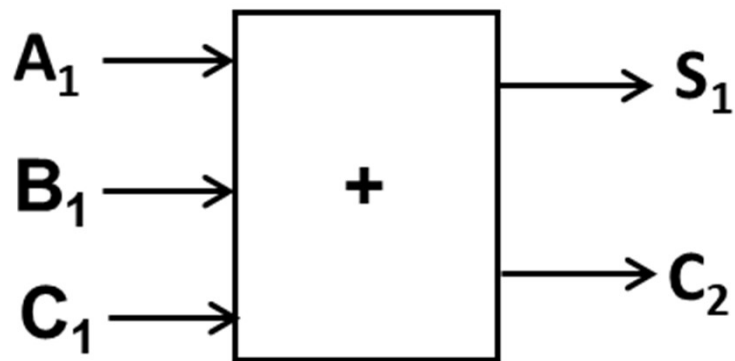
Fatia (faixa) 0:
Meio somador



Fatia (faixa) $i > 0$:
Somador completo

Somador

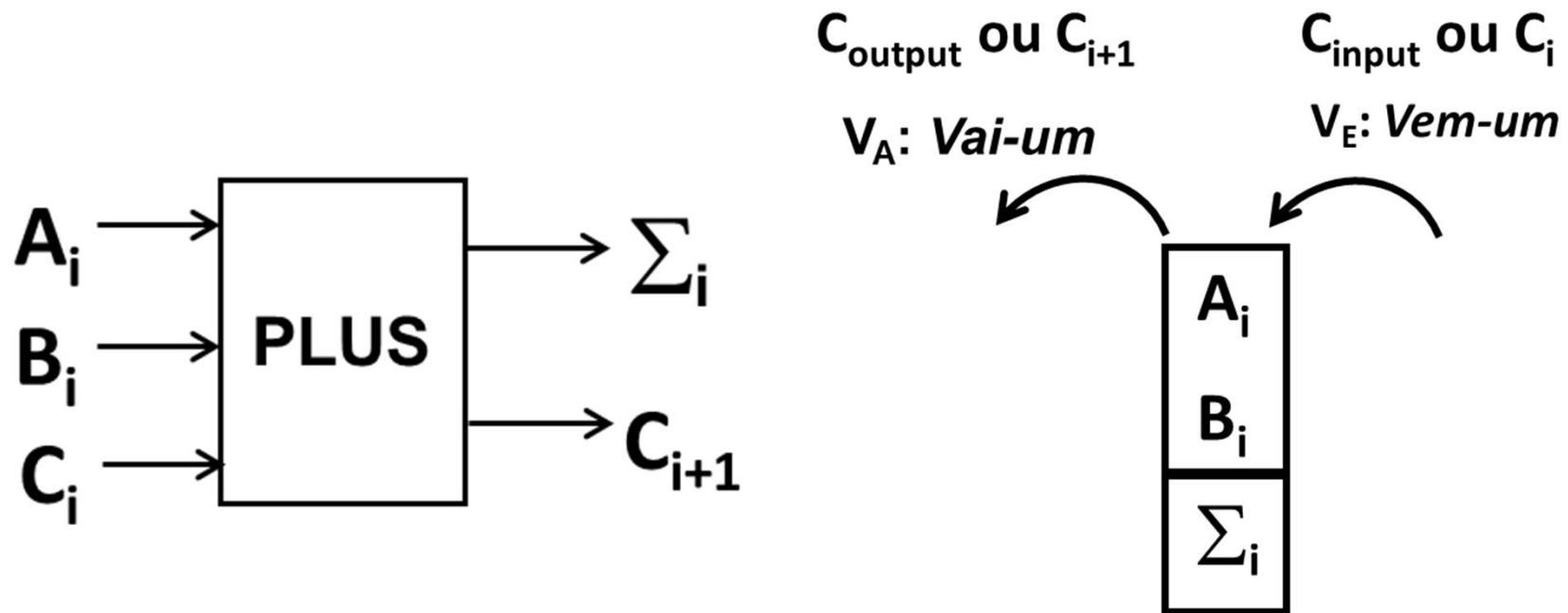
- Como construir um somador binário?
 - **Vamos adotar o módulo básico mais genérico:**
 - Definir o módulo básico;
 - Ajuste de nomenclatura – Troca do sinal de “+” por “PLUS” (adição), para não confundir com o “+” lógico (operação “OU”).



Fatia (faixa) $i \geq 0$:
Somador completo

Somador

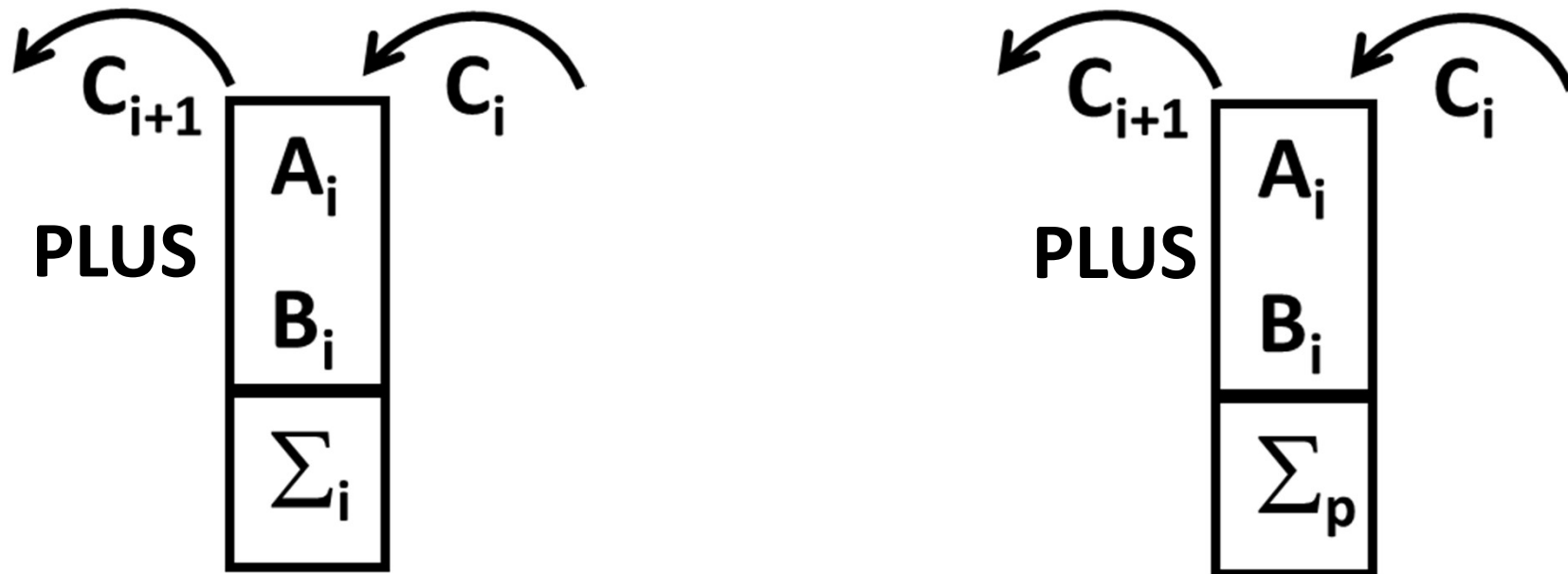
- Como construir um somador binário?
 - Projetar o módulo básico:



Somador

- Projetar o módulo básico
 - Mas o resultado final da adição na fatia (Σ_i) não pode ser obtido diretamente;
 - Primeiro obtém-se o resultado parcial (Σ_p), onde

$$\Sigma_p = A_i \text{ PLUS } B_i$$



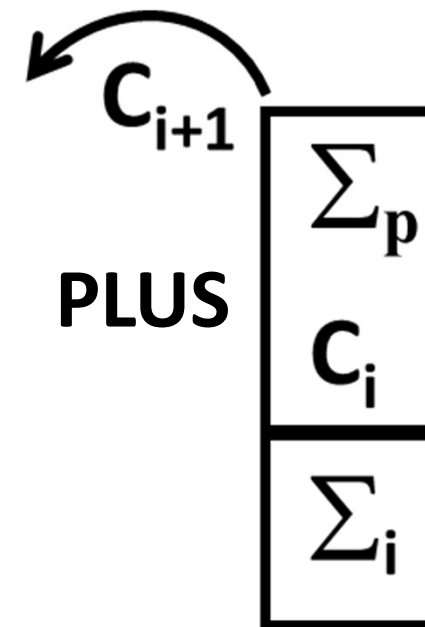
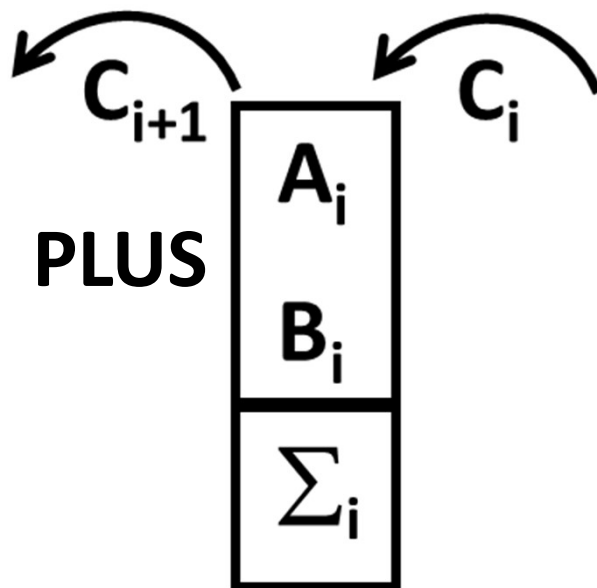
Somador

- Projetar o módulo básico

- Mas o resultado final da adição na fatia (Σ_i) não pode ser obtido diretamente;

- Depois obtém-se o resultado final (Σ_i), onde

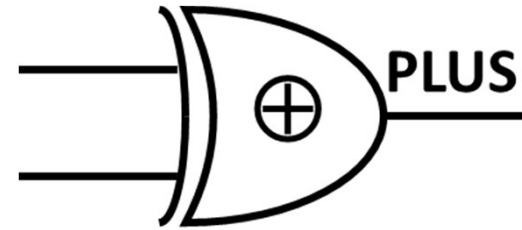
$$\Sigma_i = \Sigma_p \text{ PLUS } C_i = A_i \text{ PLUS } B_i \text{ PLUS } C_i$$



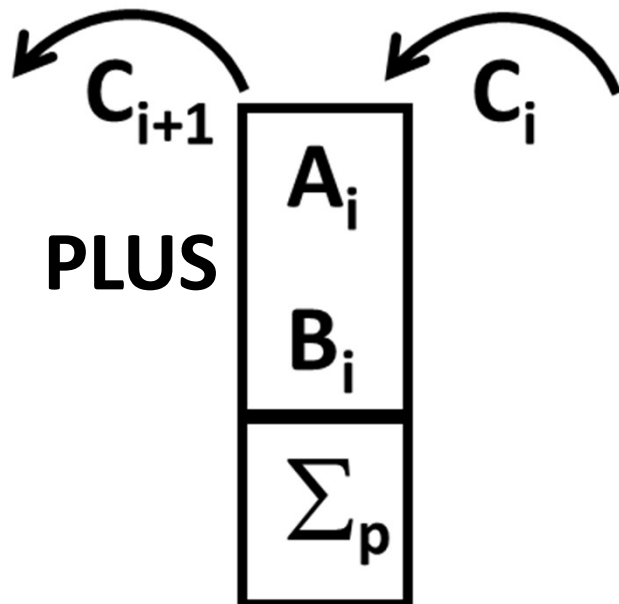
Somador

- Obtenção de Σ_p :

$$\Sigma_p = A_i \text{ PLUS } B_i$$



$$\Sigma_p = A_i \text{ XOR } B_i$$

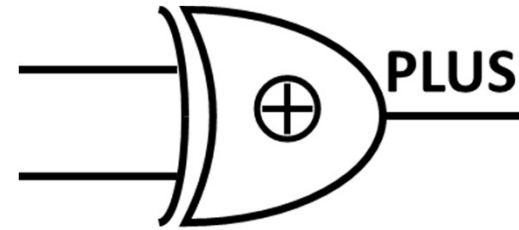


Entradas		Saida
A_i	B_i	Σ_p
0	0	0
0	1	1
1	0	1
1	1	0

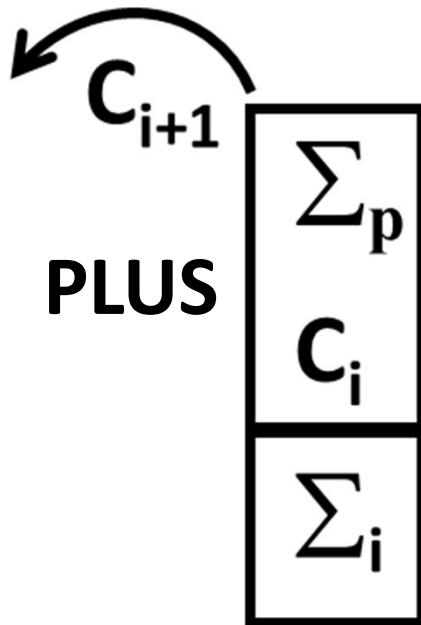
Somador

- Obtenção de Σ_i :

$$\Sigma_i = \Sigma_p \text{ PLUS } C_i$$



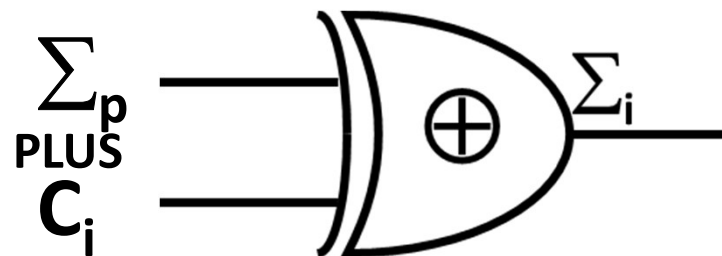
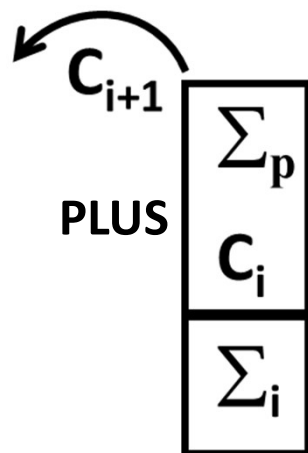
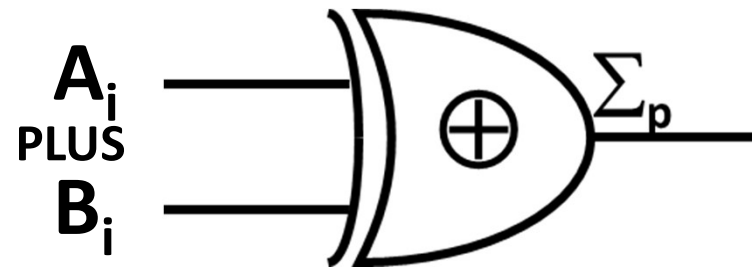
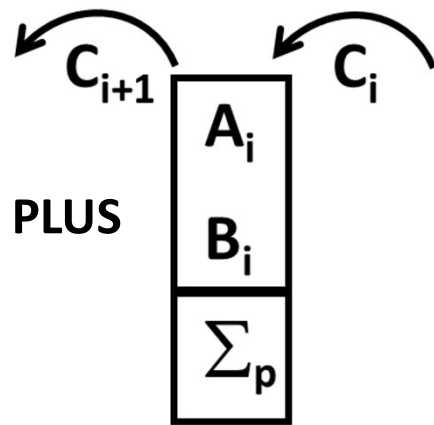
$$\Sigma_i = \Sigma_p \text{ XOR } C_i$$



Entradas		Saida
Σ_p	C_i	Σ_i
0	0	0
0	1	1
1	0	1
1	1	0

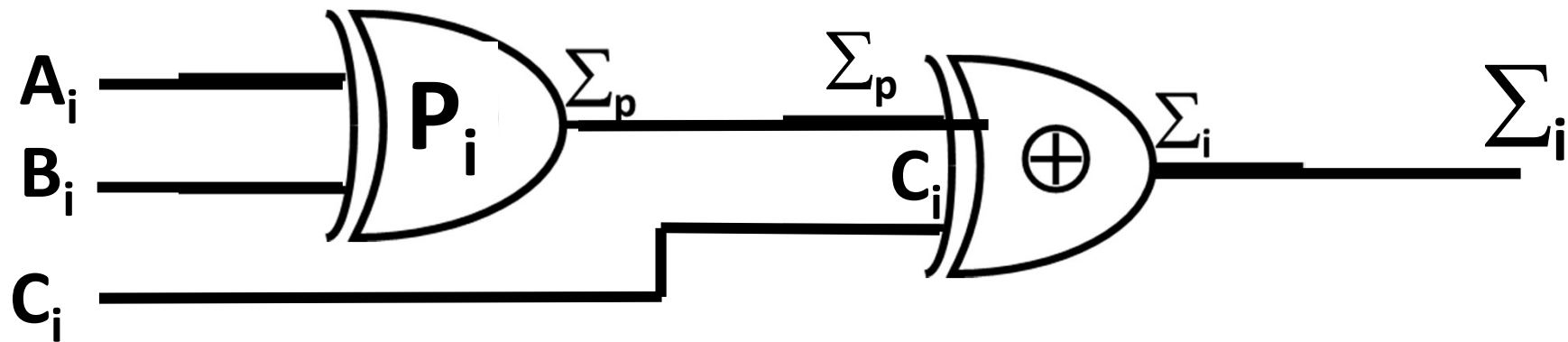
Somador

- Traduzindo a obtenção de Σ_p e Σ_i em termos de Diagrama Lógico:



Somador

- Traduzindo a obtenção de Σ_p e Σ_i em termos de Diagrama Lógico – Rearranjando o diagrama:



C_{i+1}

Somador

- Obtenção de C_{i+1} – O *carry* de saída é igual a “1” em qualquer uma de duas situações (**1-] OU 2-]**):

- **1-]** Quando ele é gerado na fatia (neste caso seu valor independe do valor de entrada C_i):

$$C_{i+1} = A_i \text{ AND } B_i = G_i = \text{Fator de Geração}$$

- **2-]** Quando os valores do *carry* de entrada (C_i) e da soma parcial (\sum_p) são ambos iguais a “1”:

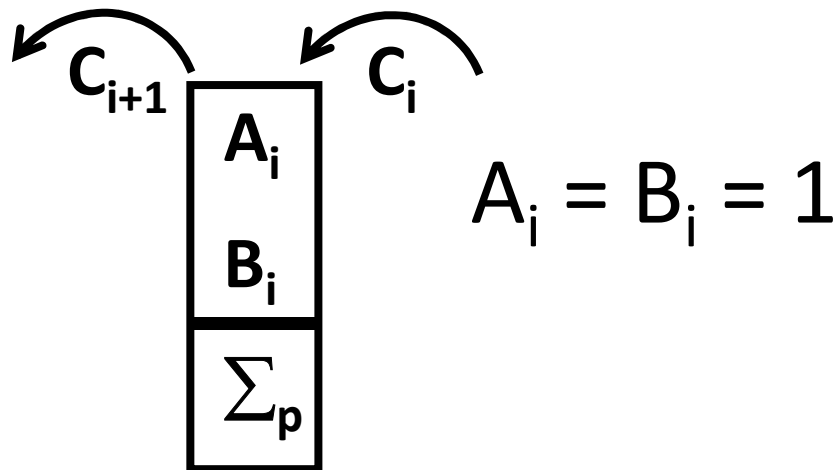
$$C_{i+1} = \sum_p \text{ AND } C_i;$$

Onde $\sum_p = P_i = \text{Fator de Propagação}$

Somador

- Obtenção de C_{i+1} – O *carry* de saída é igual a “1” em qualquer uma de duas situações (**1-] OU 2-]**):
- **1-]** Quando ele é gerado na fatia (neste caso seu valor independe do valor de entrada C_i):

$$C_{i+1} = A_i \text{ AND } B_i = G_i = \text{Fator de Geração}$$



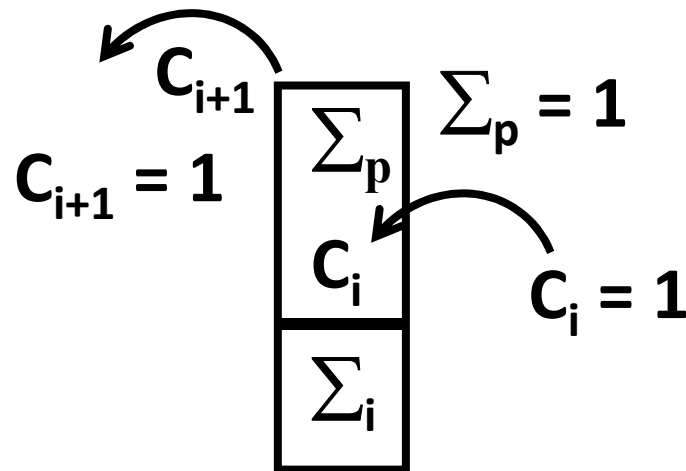
$$C_{i+1} = A_i \text{ AND } B_i = A_i$$

A_i	B_i	C_{i+1}
0	0	0
0	1	0
1	0	0
1	1	1

Somador

- Obtenção de C_{i+1} – O *carry* de saída é igual a “1” em duas situações: 2-] Quando os valores do *carry* de entrada (C_i) e da soma parcial (Σ_p) são ambos iguais a “1”: $C_{i+1} = \Sigma_p \text{ AND } C_i$;

Onde $\Sigma_p = P_i = \text{Fator de Propagação}$



$$C_{i+1} = \Sigma_p \text{ AND } C_i$$

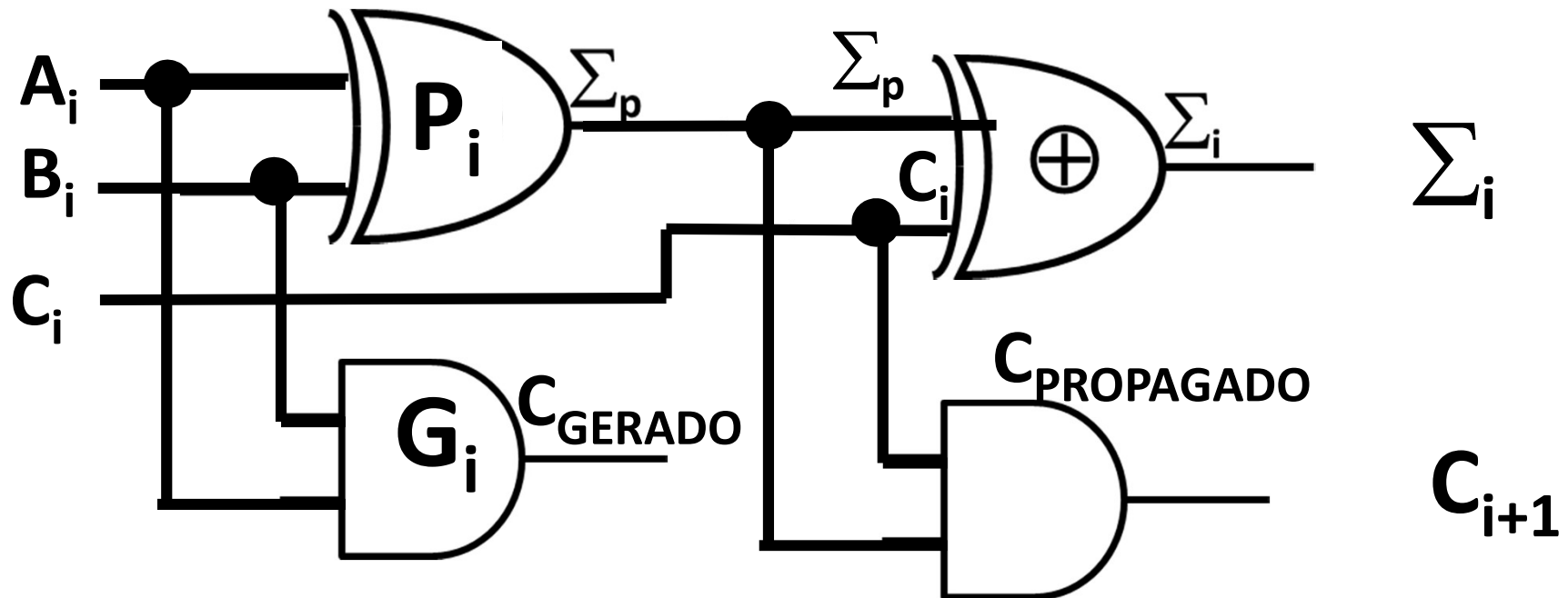
$\Sigma_p = P_i$
 $P_i =$
 Fator de Propagação

Σ_p	C_i	C_{i+1}
0	0	0
0	1	0
1	0	0
1	1	1

Somador

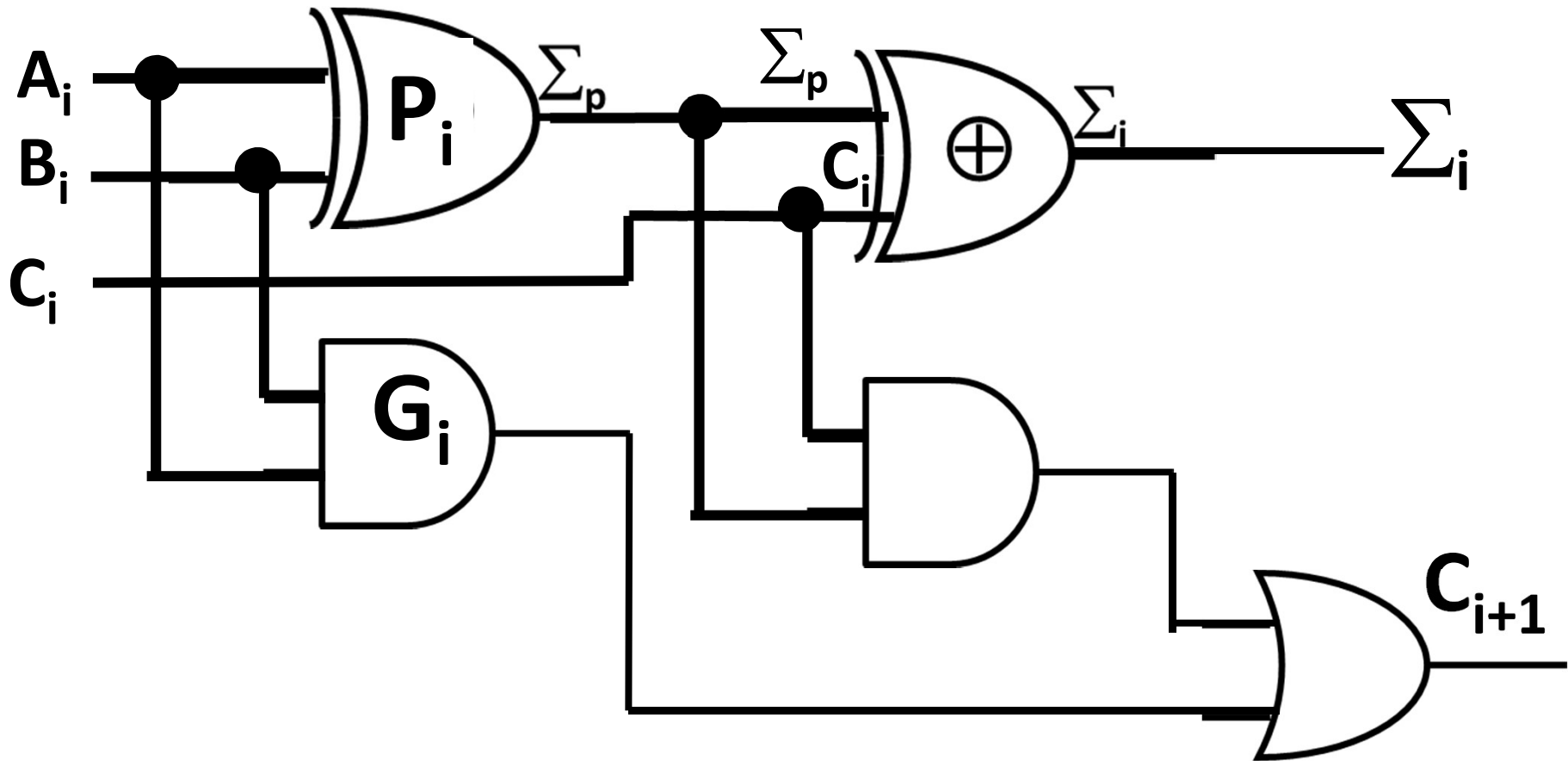
- Incorporando a obtenção de C_{i+1} às de Σ_p e Σ_i
– Rearranjando o diagrama lógico:

Fator de Propagação = $P_i = A_i \text{ XOR } B_i$

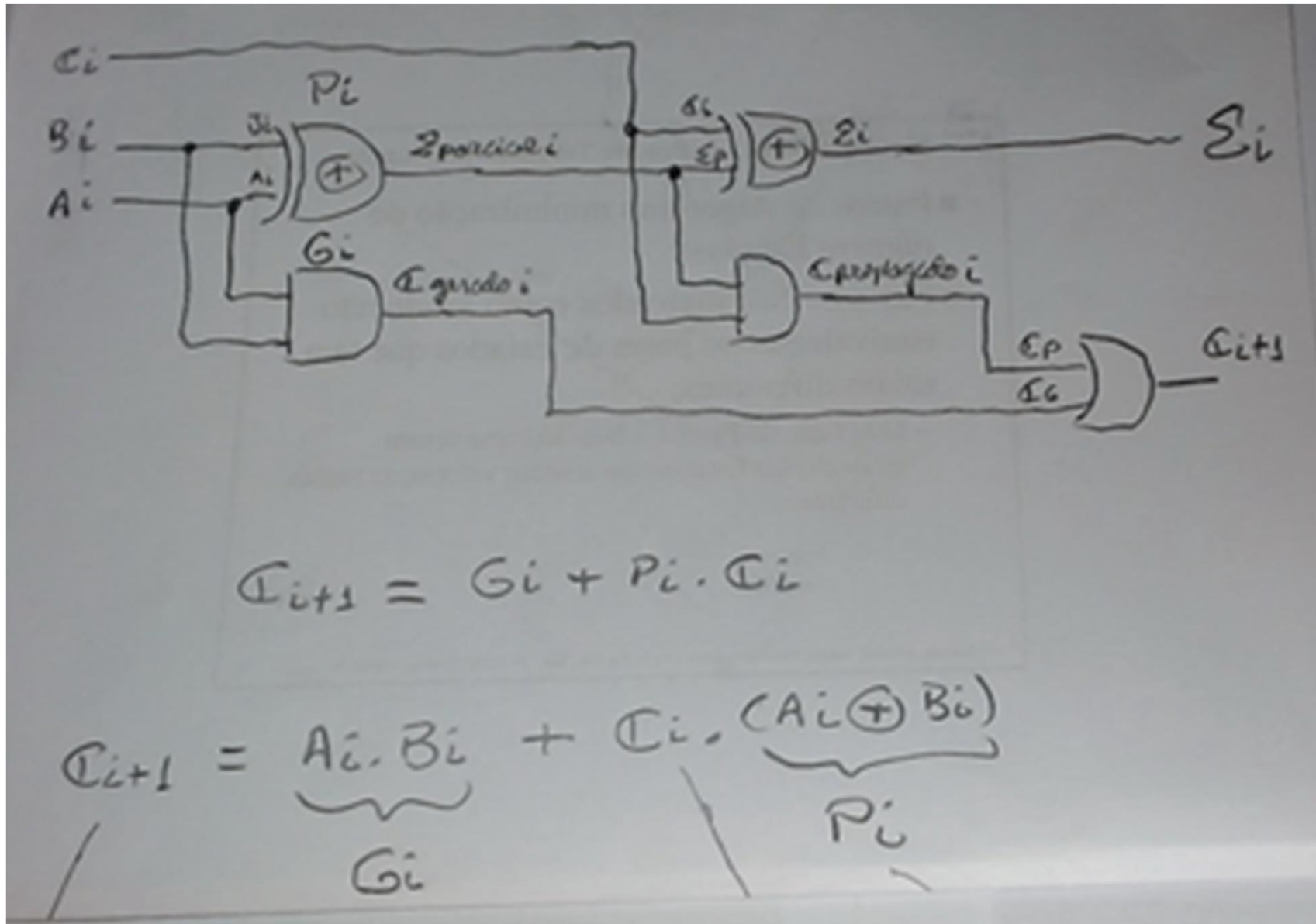


Fator de Geração = $G_i = A_i \text{ AND } B_i$

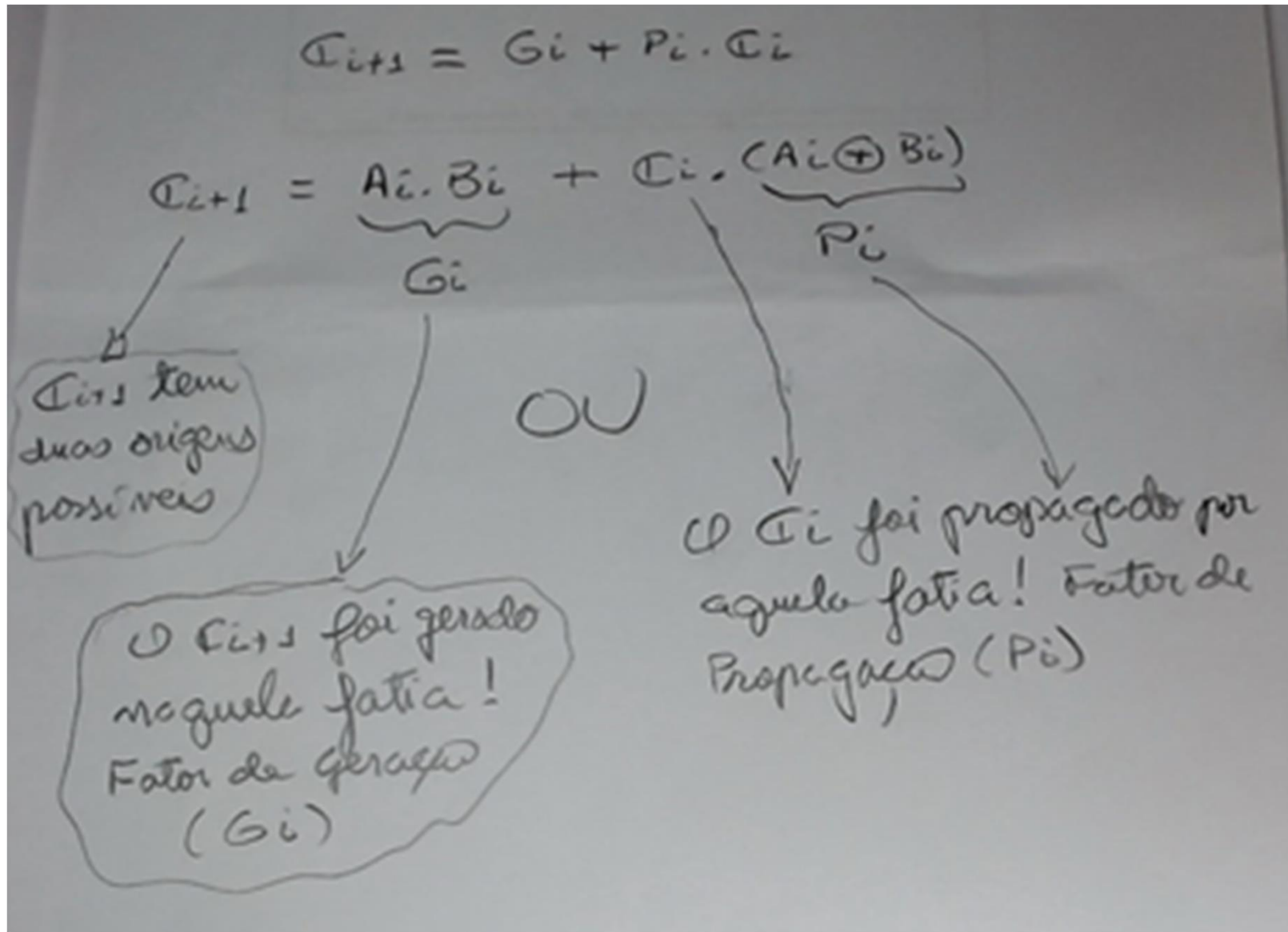
Somador Completo: Carry_{Gerado} OU Carry_{Propagado}



C_{i+1} – Gerado na fatia “i” ou Propagado por esta

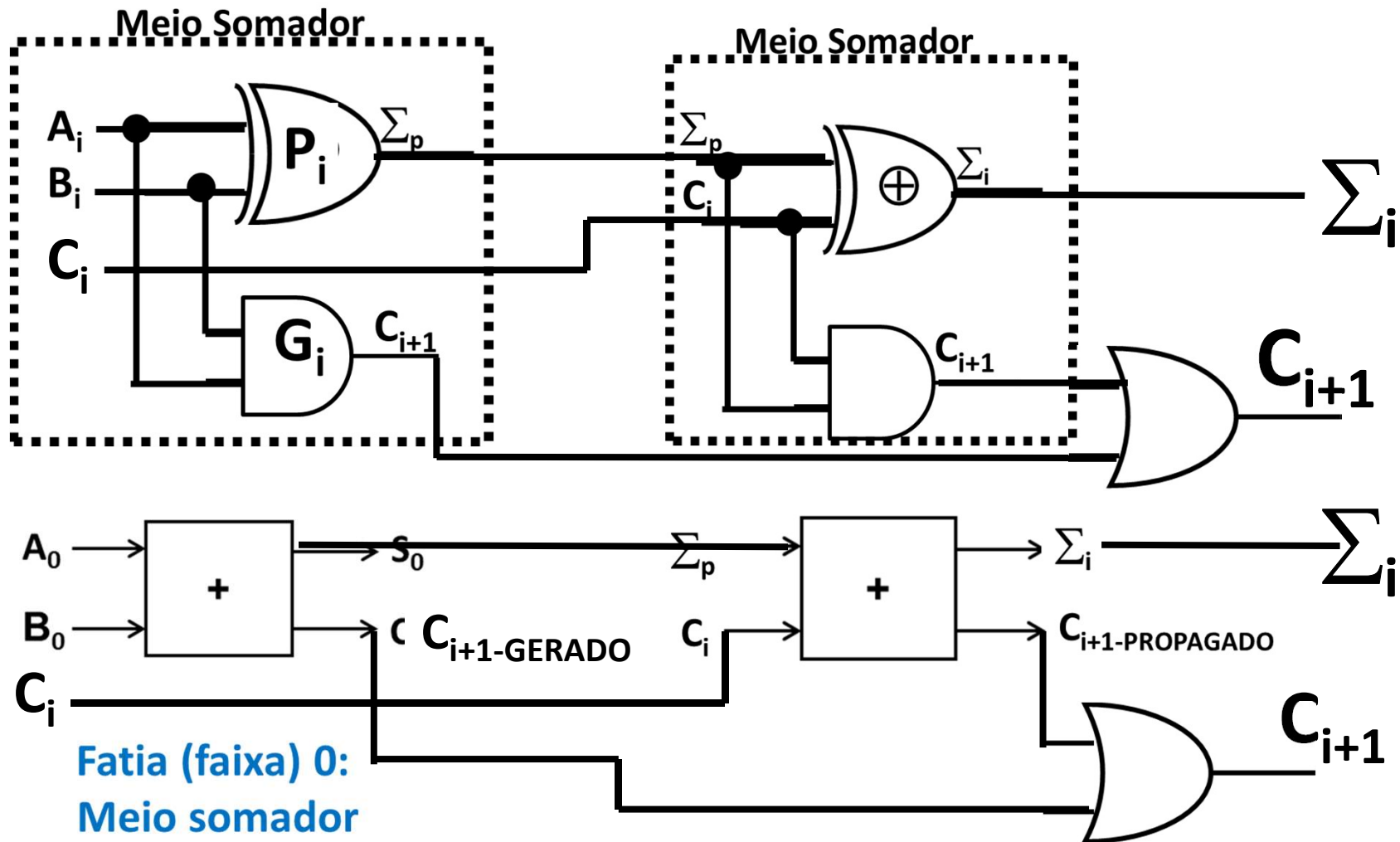


C_{i+1} – Gerado na fatia “i” ou Propagado por esta



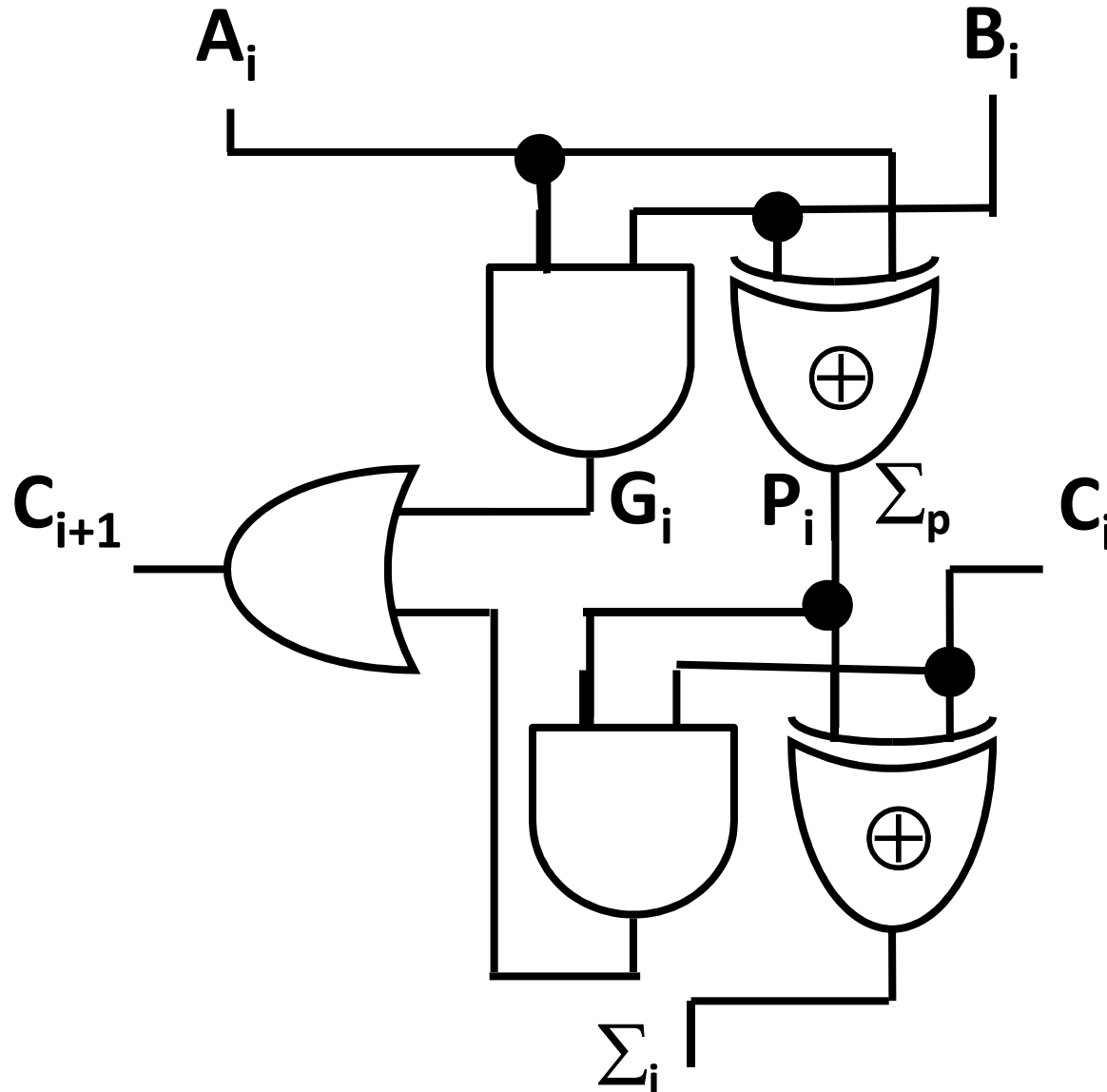
Somador

- Cascateando 2 módulos do tipo “Meio Somador”, para compor 1 módulo “Somador Completo”:



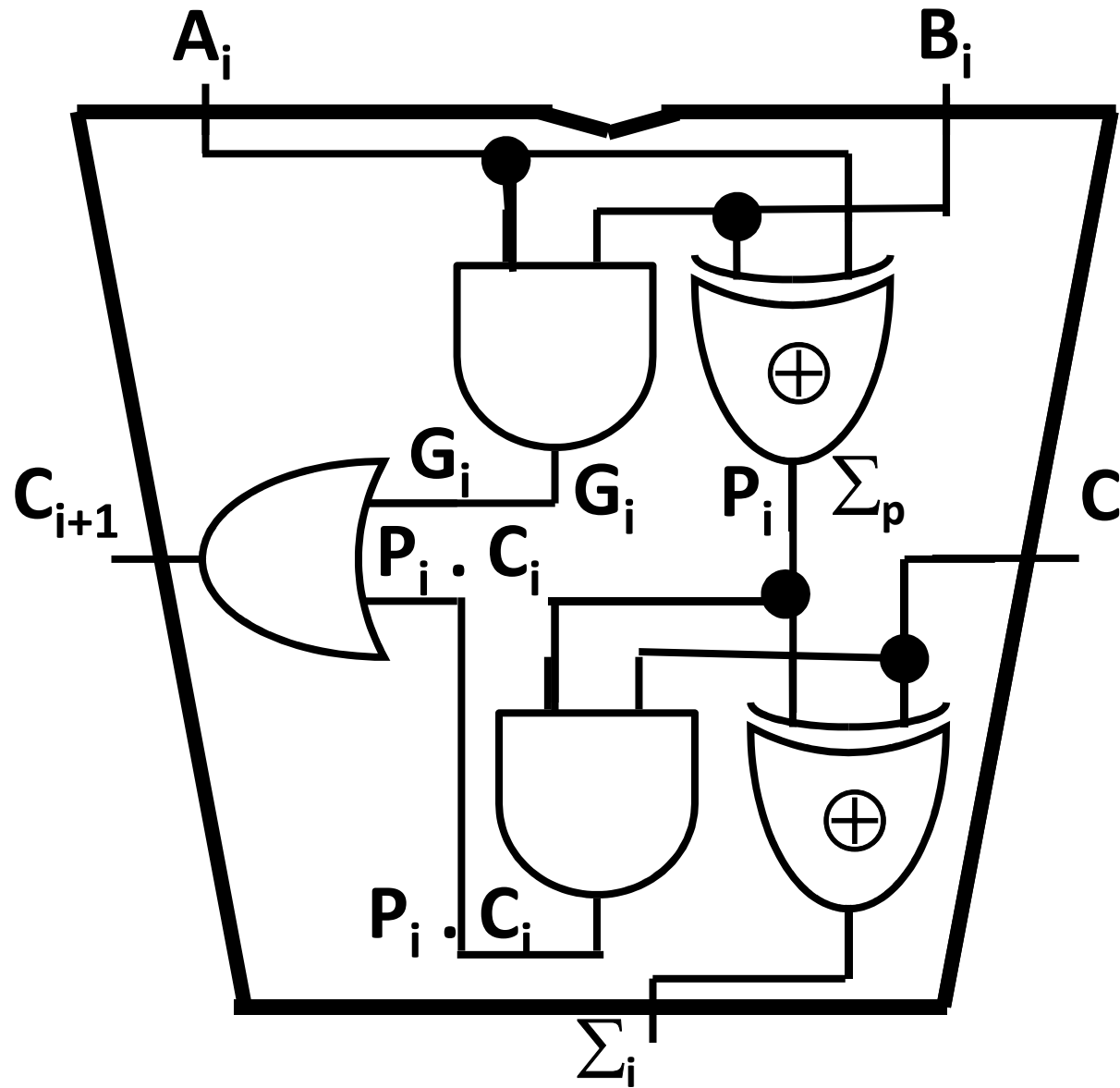
Somador Rearranjado de duas formas, nos será útil mais adiante.

Formato [1]



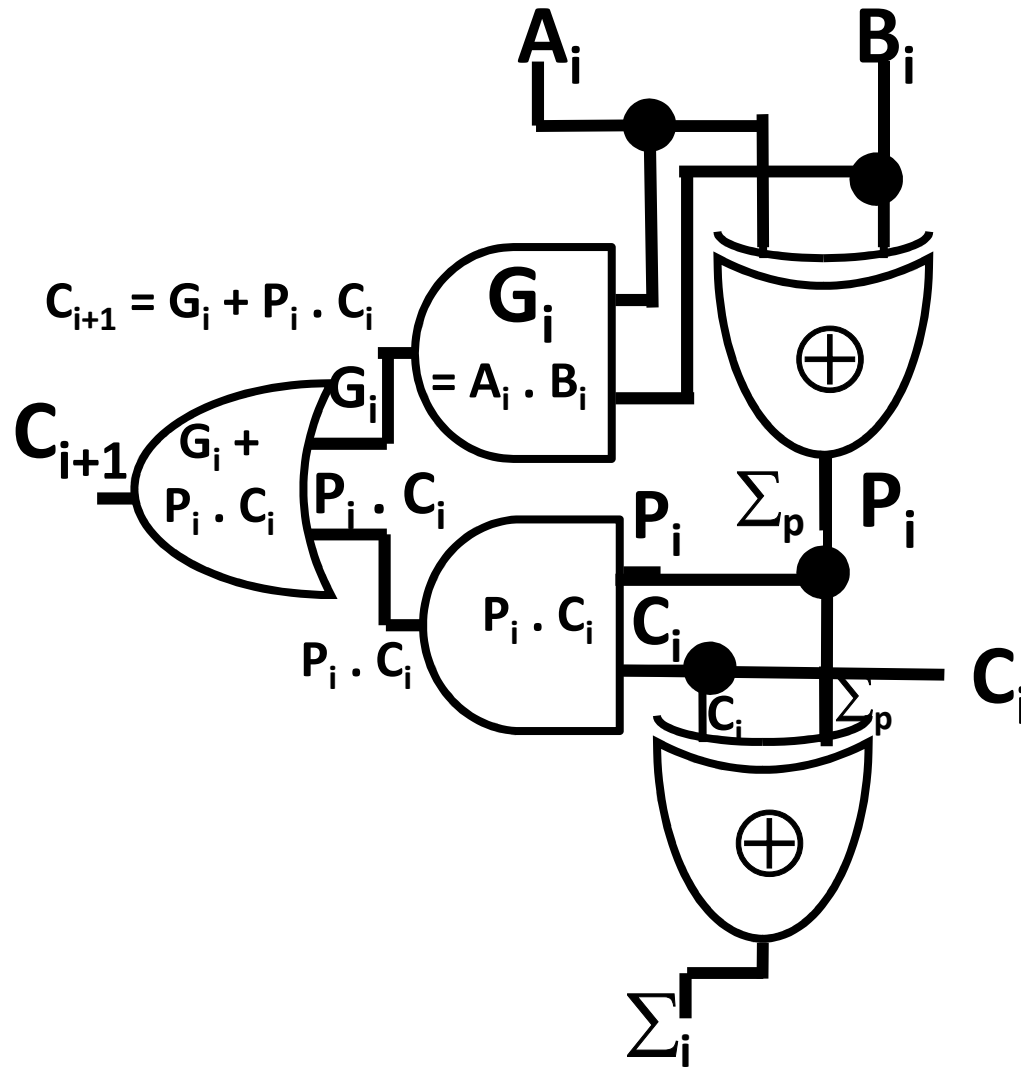
Somador

Formato [1]



Somador Rearranjado de duas formas, nos será útil mais adiante.

Formato [2]



Somador

Formato [2]

