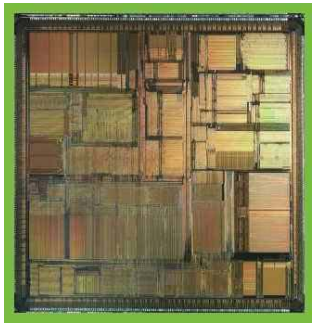
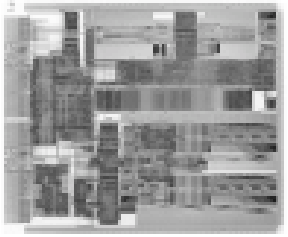
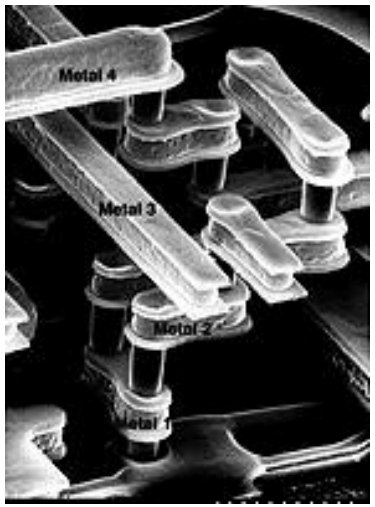




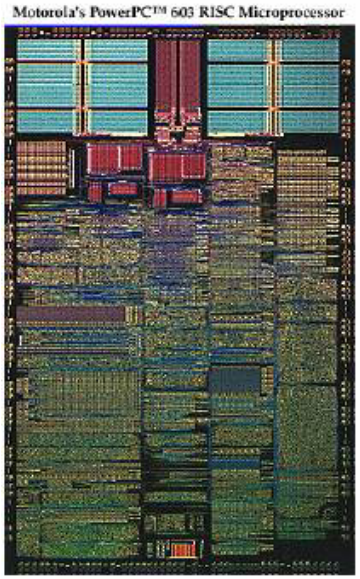
VLSI Arithmetic



Lecture 10:
Multipliers

Prof. Vojin G. Oklobdzija
University of California

http://www.ece.ucdavis.edu/acsel



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Multiplication

Algorithm:

$$P = XY = X \times \sum_{i=0}^{n-1} y^i r^i = \sum_{i=0}^{n-1} X \times y^i r^i$$

$$p^{(0)} = 0 \qquad \textit{initially}$$

$$p^{(j+1)} = \frac{1}{r}(p^j + r^n X y_j) \quad \textit{for } j=0,\ldots,n\text{-}1$$

$$p(n) = XY \quad \textit{after } n \textit{ steps}$$

Multiplication Algorithm*

Notation for our discussion of multiplication algorithms:

a	Multiplicand	$a_{k-1}a_{k-2} \cdots a_1a_0$
x	Multiplier	$x_{k-1}x_{k-2} \cdots x_1x_0$
p	Product ($a \times x$)	$p_{2k-1}p_{2k-2} \cdots p_1p_0$

Initially, we assume unsigned operands

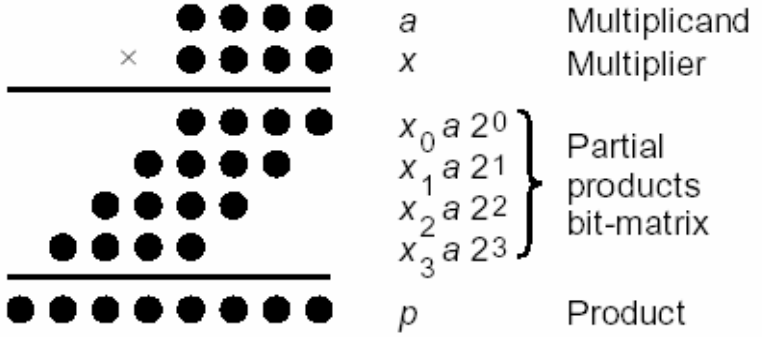


Fig. 9.1 Multiplication of two 4-bit unsigned binary numbers in dot notation.

**from Parhami*

Multiplication Algorithm*

Multiplication with right shifts: top-to-bottom accumulation

$p^{(j+1)} = (p^{(j)} + x_j a 2^k) 2^{-1}$ with $p^{(0)} = 0$ and

|——add——|

|—shift right—|

$p^{(k)} = p = ax + p^{(0)} 2^{-k}$

Multiplication with left shifts: bottom-to-top accumulation

$p^{(j+1)} = 2p^{(j)} + x_{k-j-1} a$ with $p^{(0)} = 0$ and

|shift|

|——add——|

$p^{(k)} = p = ax + p^{(0)} 2^k$

**from Parhami*

Multiplication Algorithm*

Right-shift algorithm

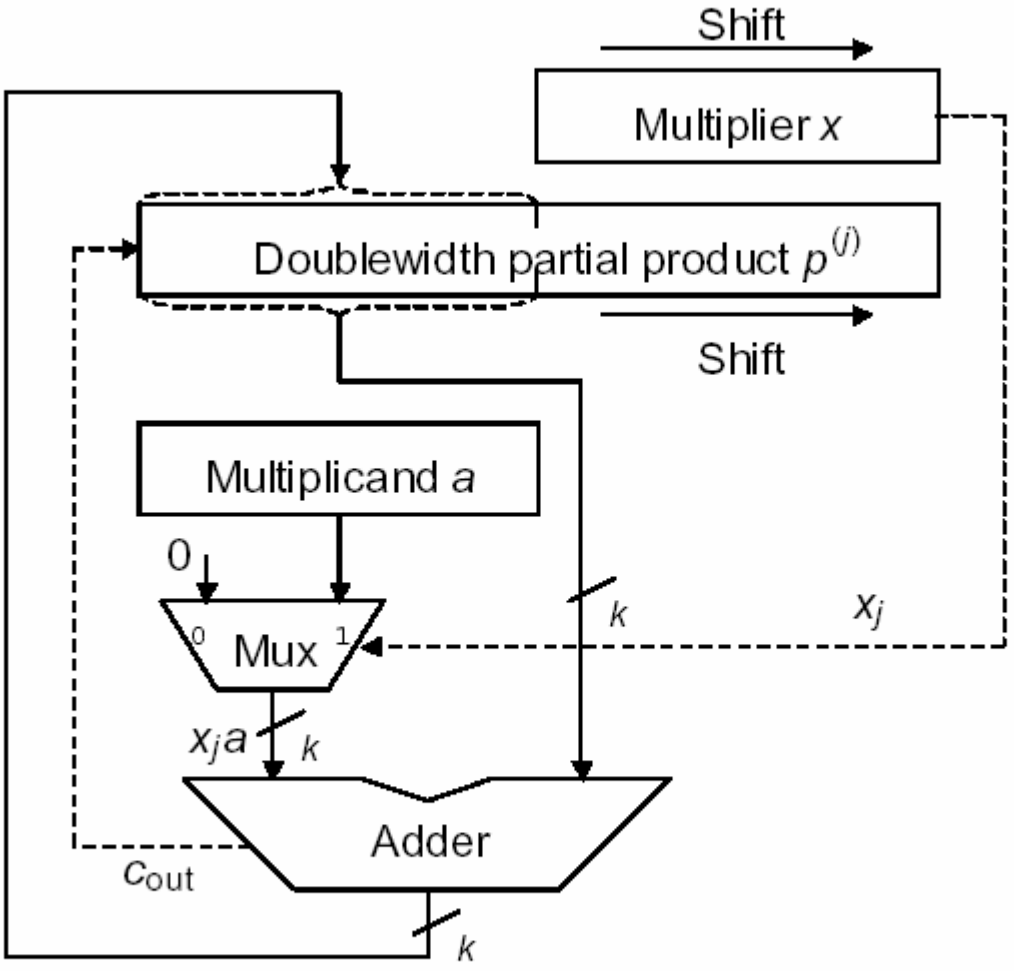
a			1	0	1	0		
x			1	0	1	1		
$p^{(0)}$			0	0	0	0		
$+x_0a$			1	0	1	0		
$2p^{(1)}$	0	1	0	1	0			
$p^{(1)}$			0	1	0	1	0	
$+x_1a$			1	0	1	0		
$2p^{(2)}$	0	1	1	1	1	0		
$p^{(2)}$			0	1	1	1	1	0
$+x_2a$			0	0	0	0		
$2p^{(3)}$	0	0	1	1	1	1	0	
$p^{(3)}$			0	0	1	1	1	1	0
$+x_3a$			1	0	1	0		
$2p^{(4)}$	0	1	1	0	1	1	1	0
$p^{(4)}$			0	1	1	0	1	1	1	0

Left-shift algorithm

a					1	0	1	0
x					1	0	1	1
$p^{(0)}$					0	0	0	0
$2p^{(0)}$			0		0	0	0	0
$+x_3a$					1	0	1	0
$p^{(1)}$				0	1	0	1	0
$2p^{(1)}$			0	1	0	1	0	0
$+x_2a$					0	0	0	0
$p^{(2)}$			0	1	0	1	0	0
$2p^{(2)}$		0	1	0	1	0	0	0
$+x_1a$					1	0	1	0
$p^{(3)}$		0	1	1	0	0	1	0
$2p^{(3)}$	0	1	1	0	0	1	0	0
$+x_0a$					1	0	1	0
$p^{(4)}$	0	1	1	0	1	1	1	0

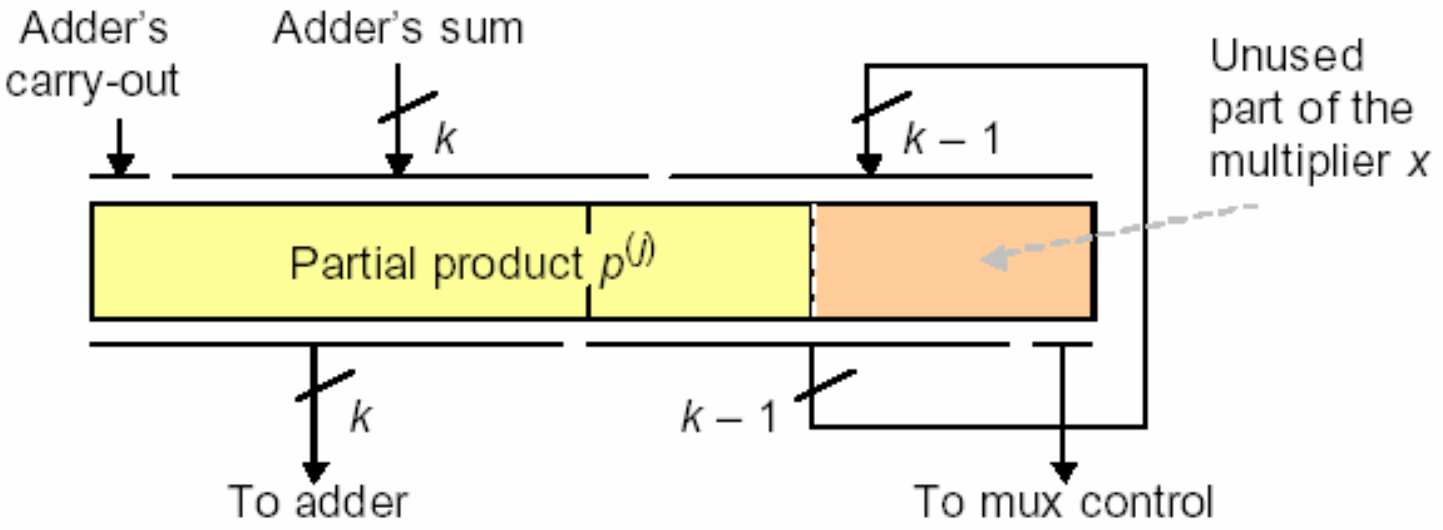
*from Parhami

Basic Hardware Multipliers



Hardware realization of the sequential multiplication algorithm with additions and right shifts. *from Parhami

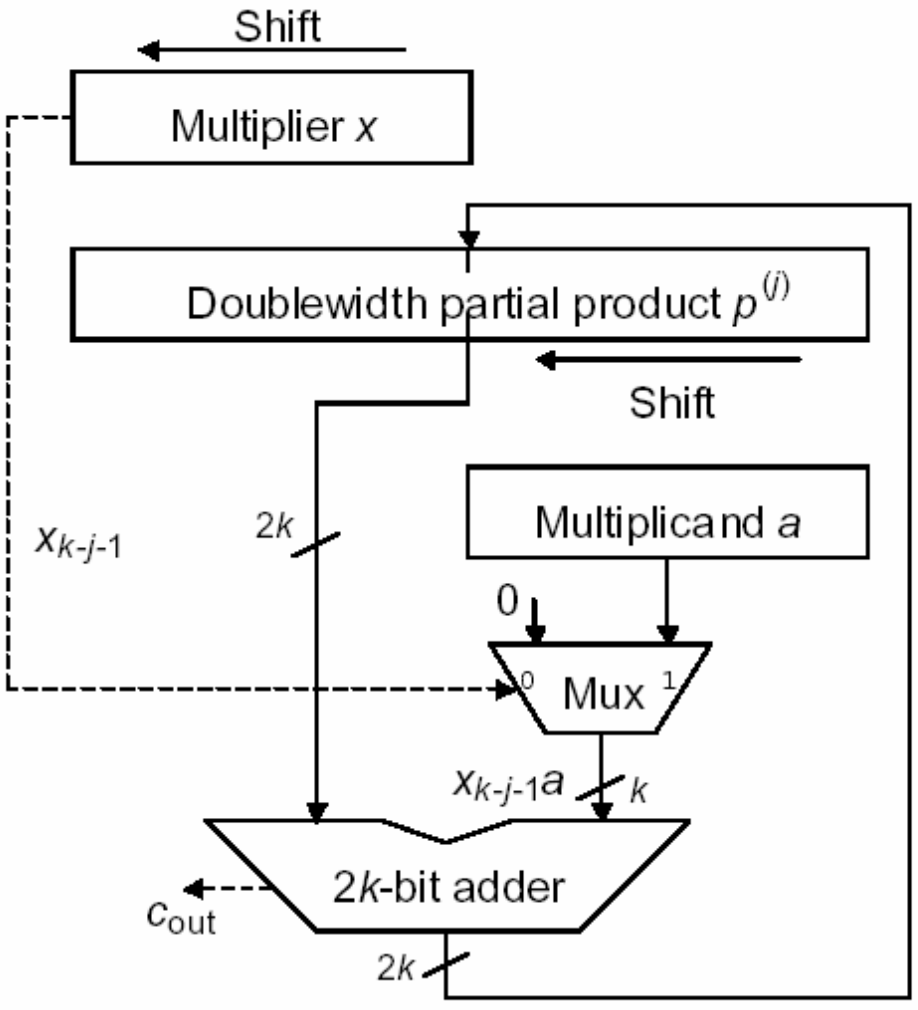
Multiplication*



Combining the loading and shifting of the double-width register holding the partial product and the partially used multiplier.

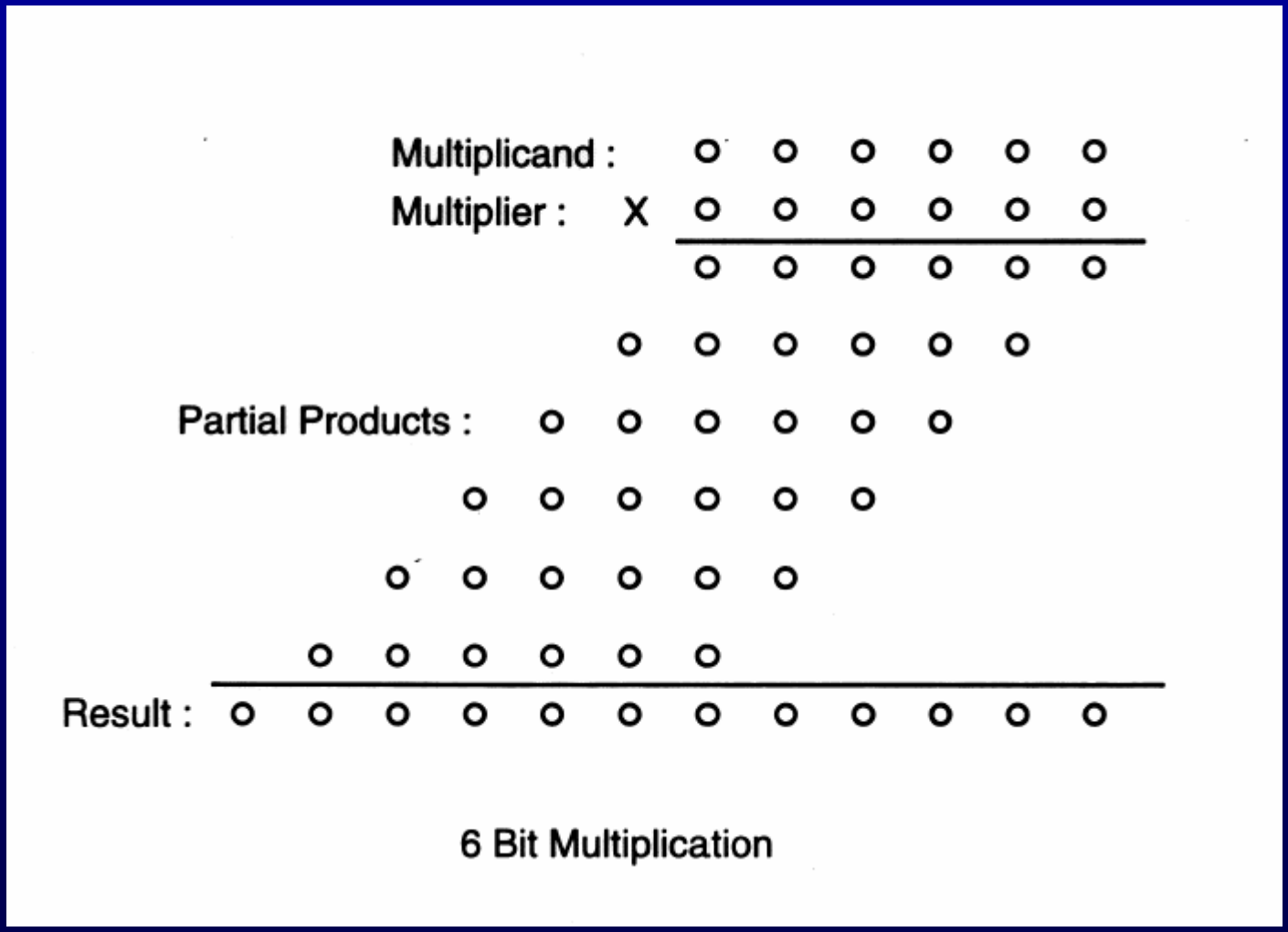
**from Parhami*

Multiplication*



Hardware realization of the sequential multiplication algorithm with left shifts and additions.

**from Parhami*



6 Bit Multiplication

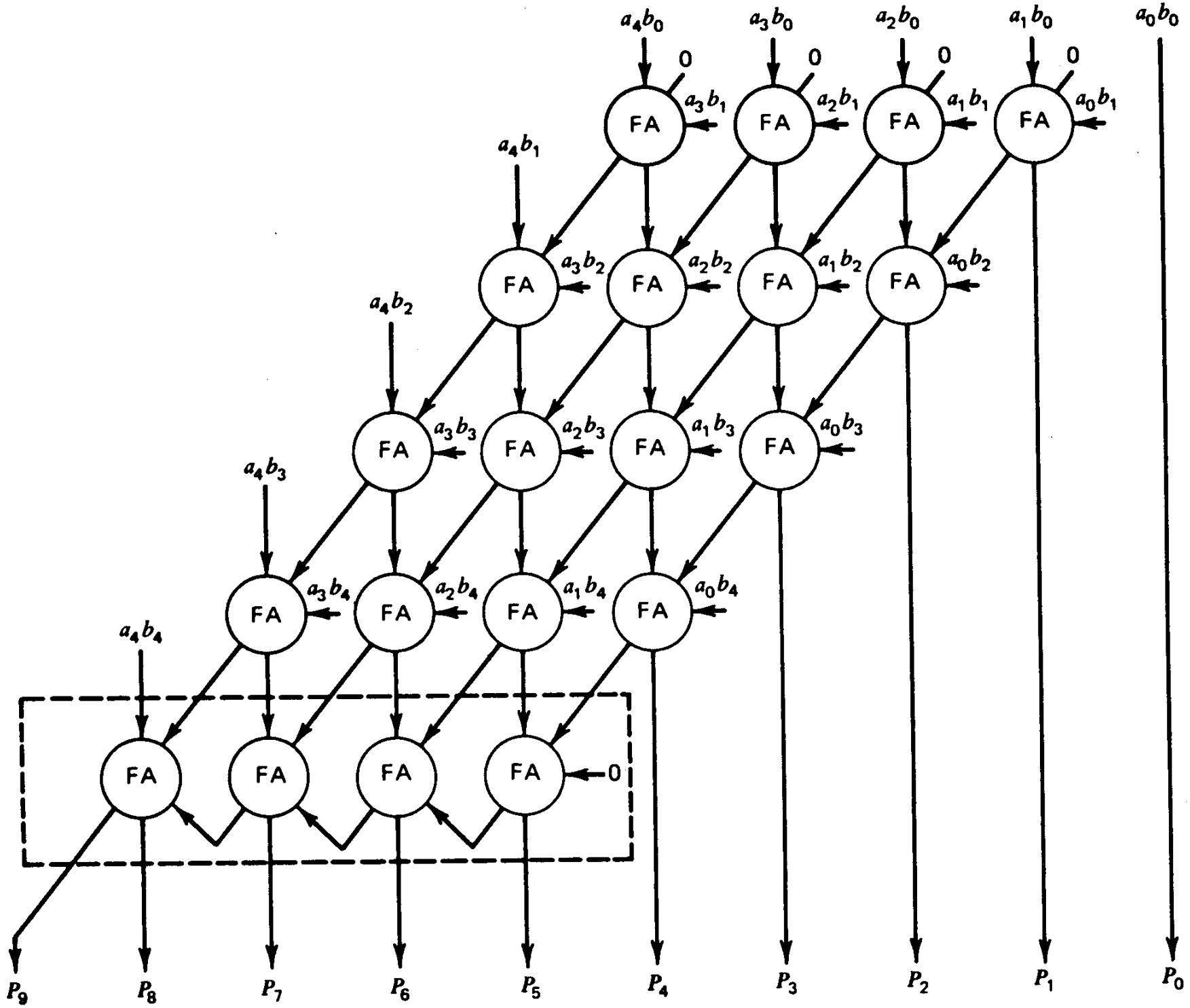


Figure 6.3 The schematic circuit diagram of a 5-by-5 unsigned array multiplier (Braun [5]).

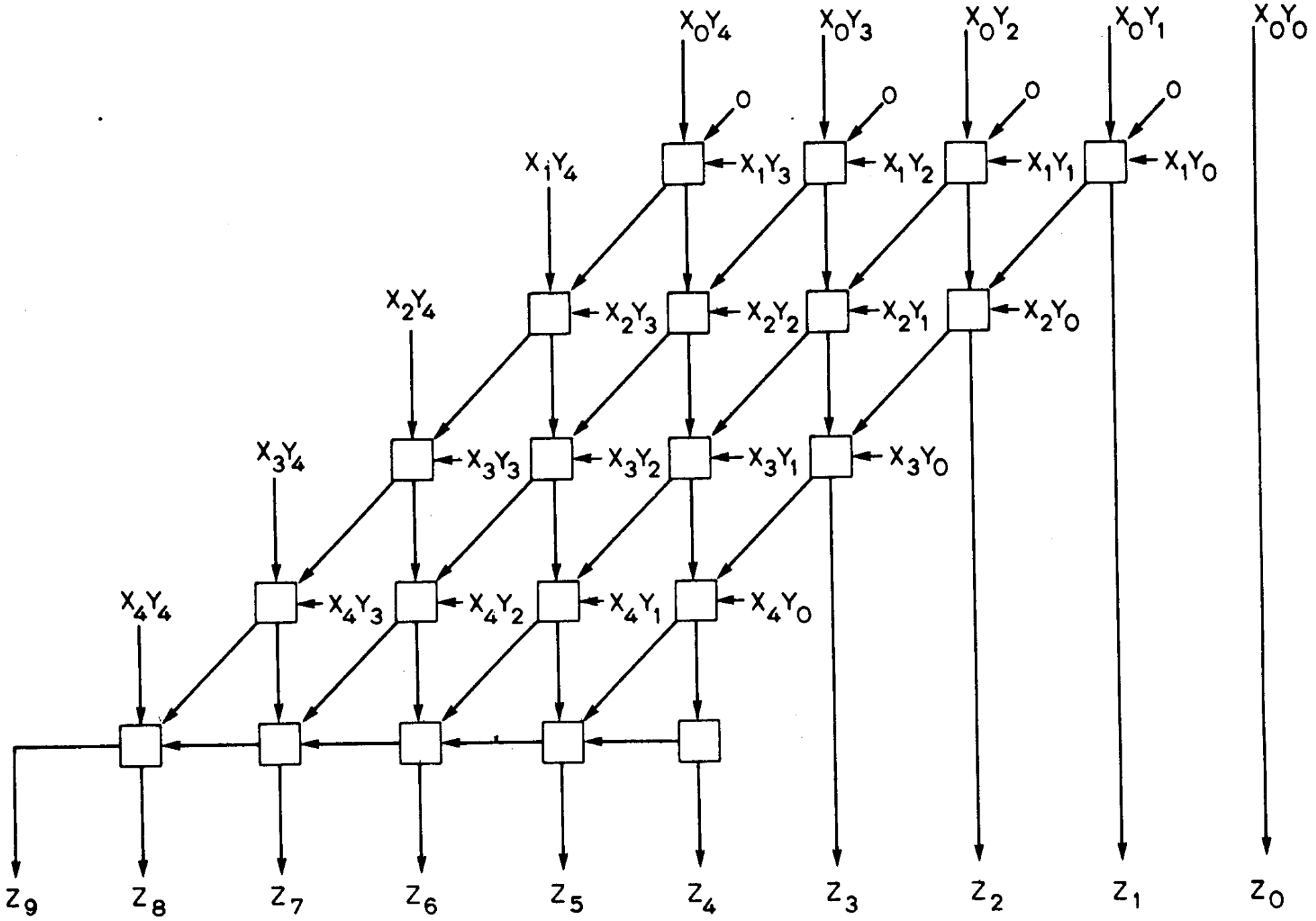


Figure 4–15a. 5×5 unsigned multiplication.

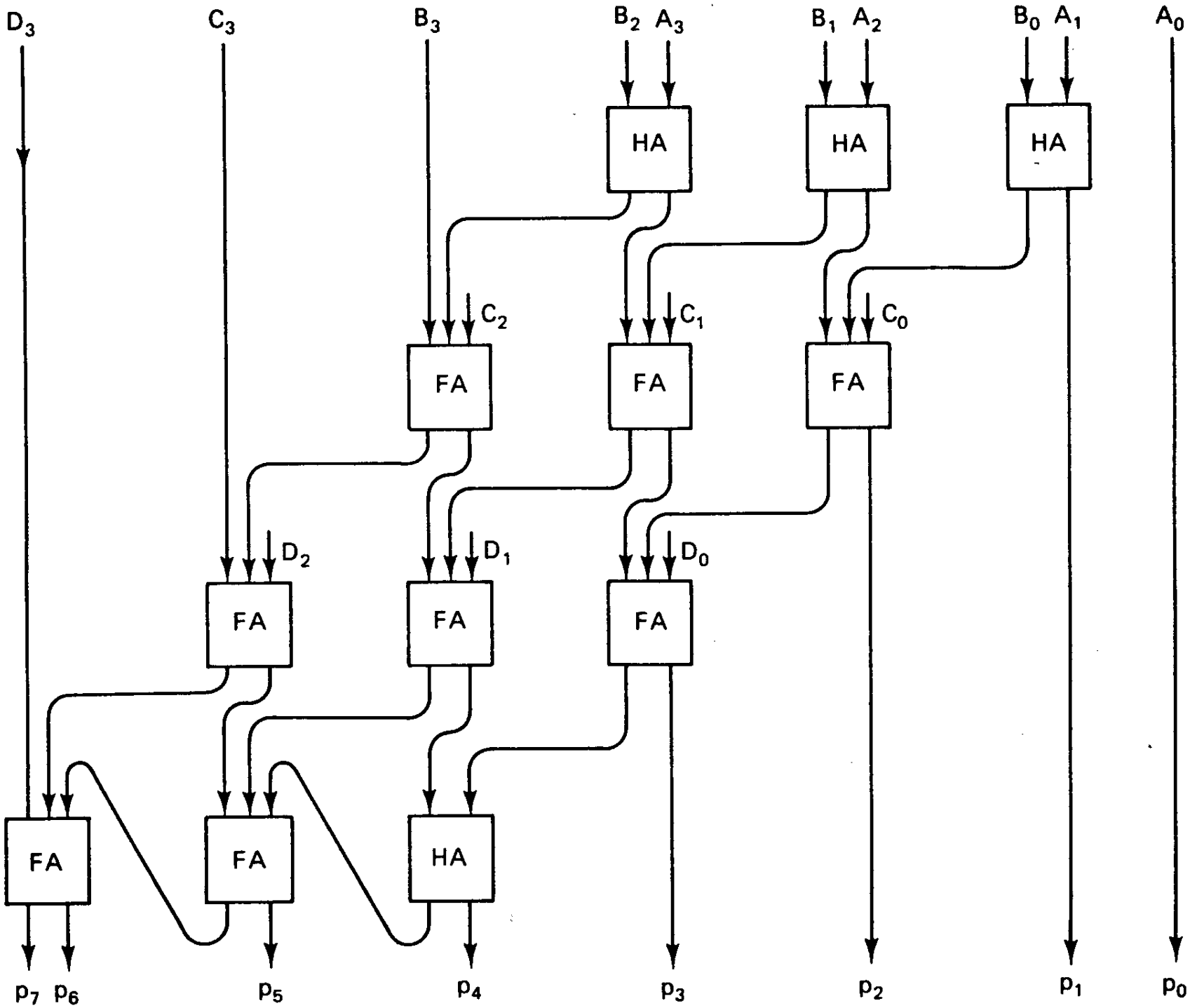


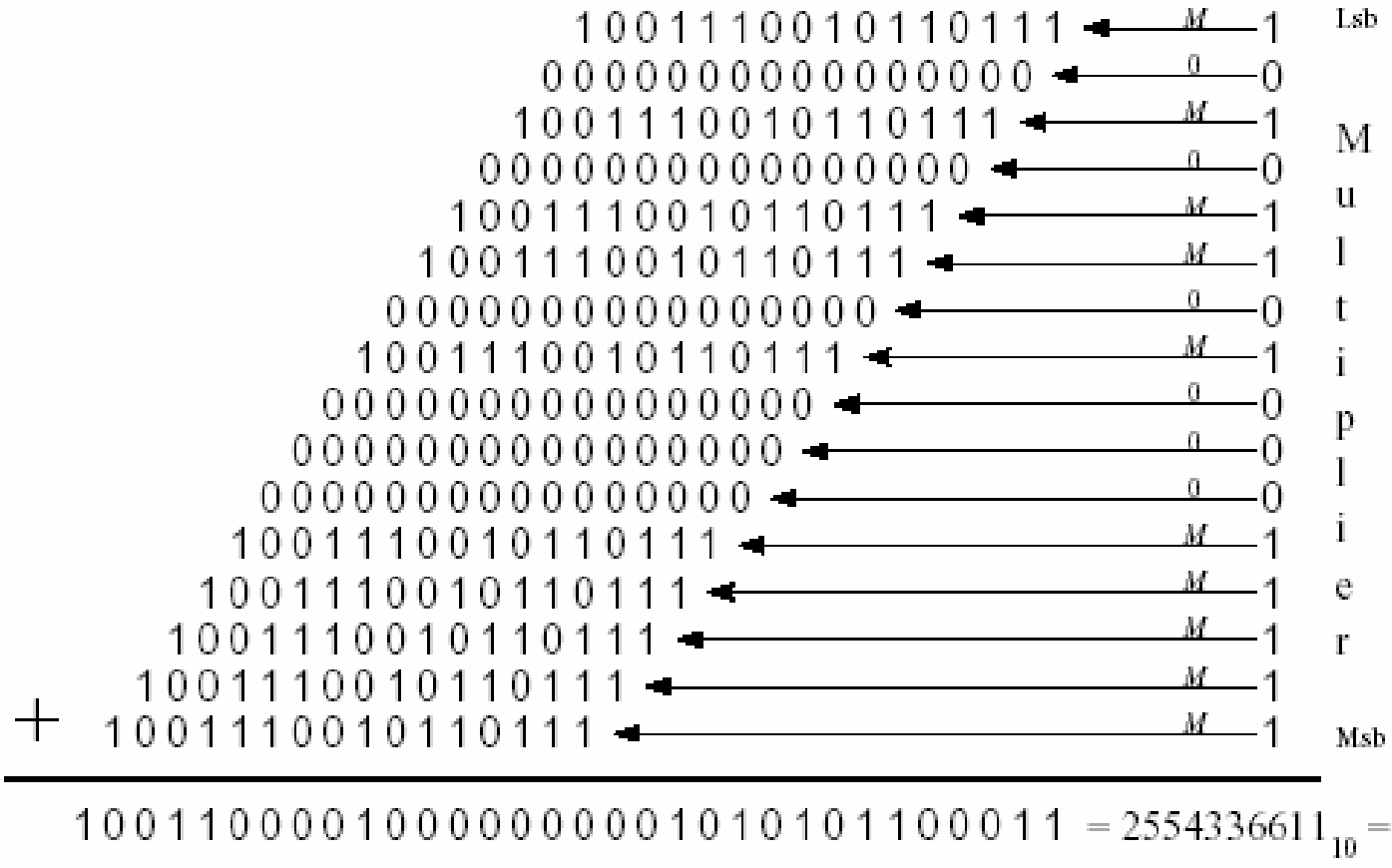
Figure 4.4 Array of half-adders and full-adders for 4-bit $\times$ 4-bit multiplication.

Generating Partial Products

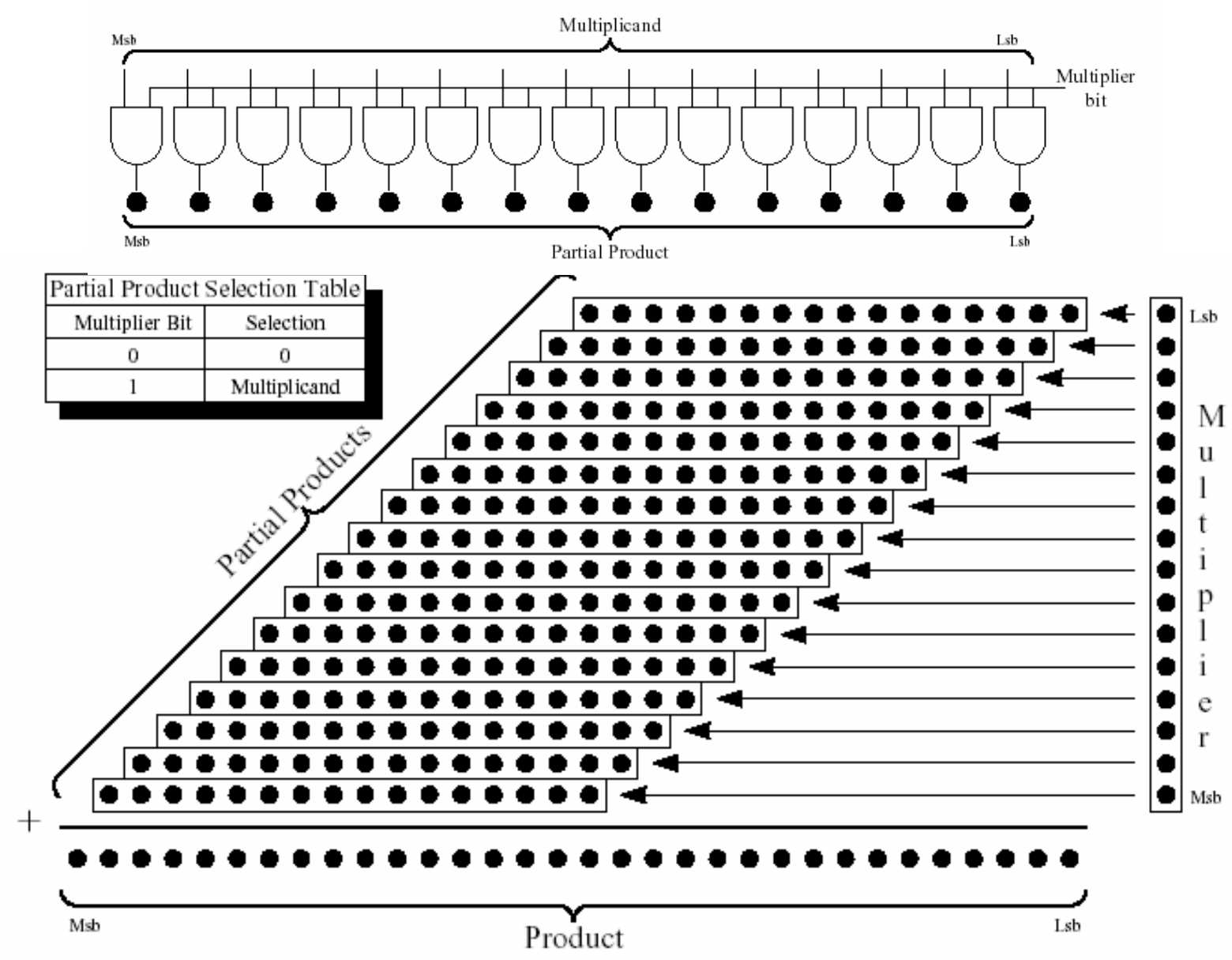
*from G. Bewick

Multiplier = 63669_{10} = 1111100010110101

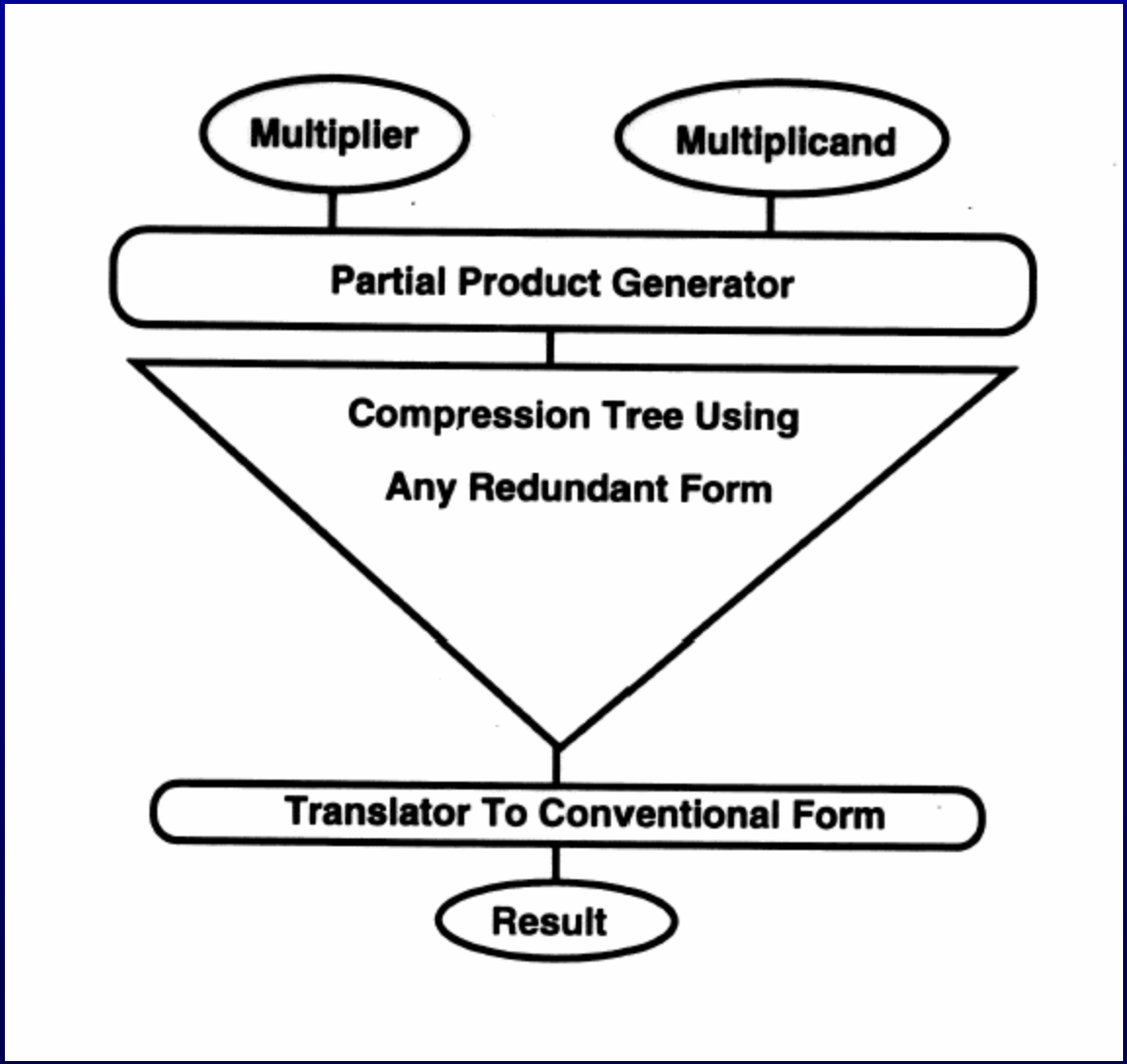
Multiplicand (M) = 40119_{10} = 1001110010110111



Generating Partial Products



*from G. Bewick


Multiplier
Multiplicand
Partial Product Generator
Compression Tree Using
Any Redundant Form
Translator To Conventional Form
Result

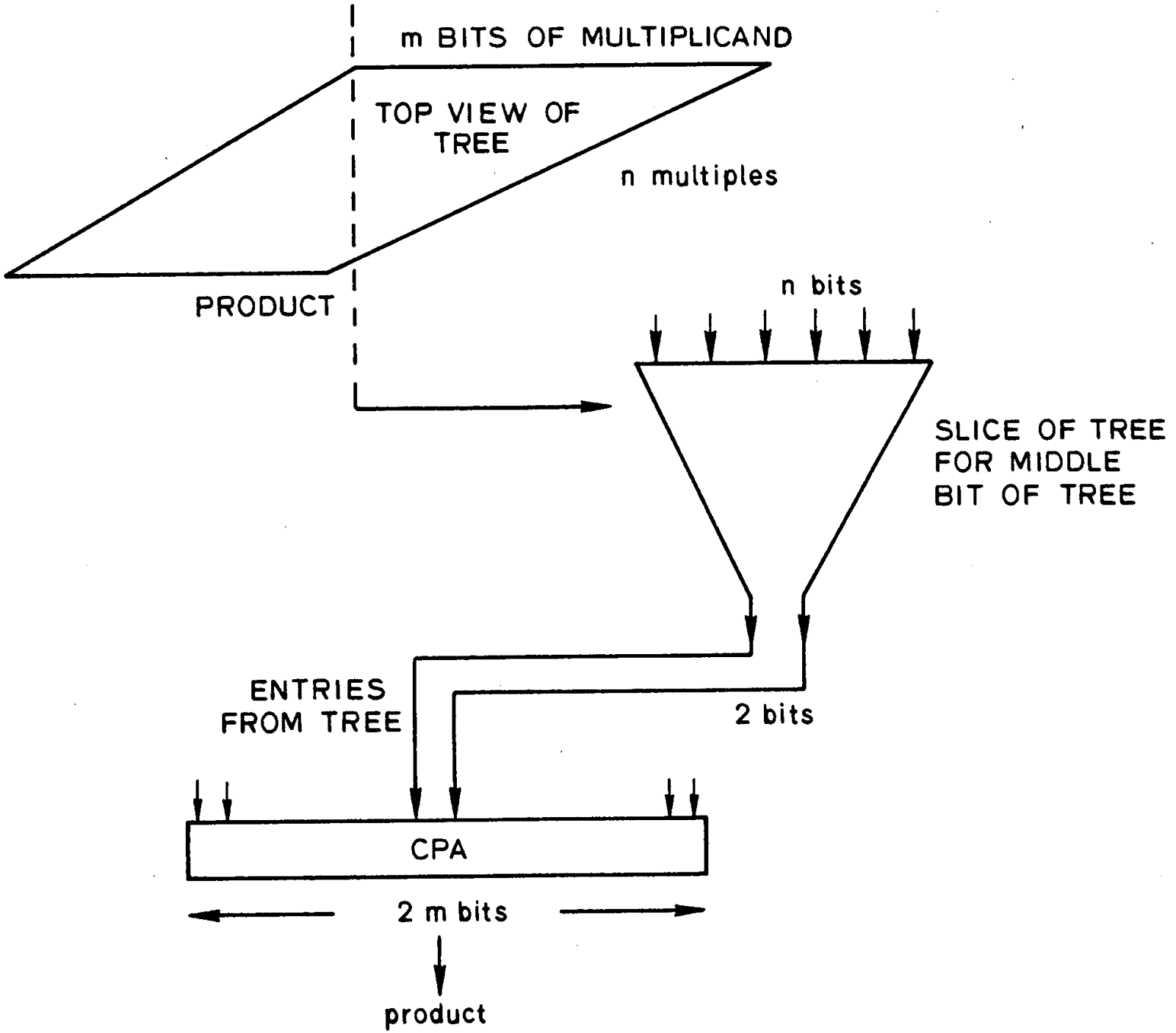
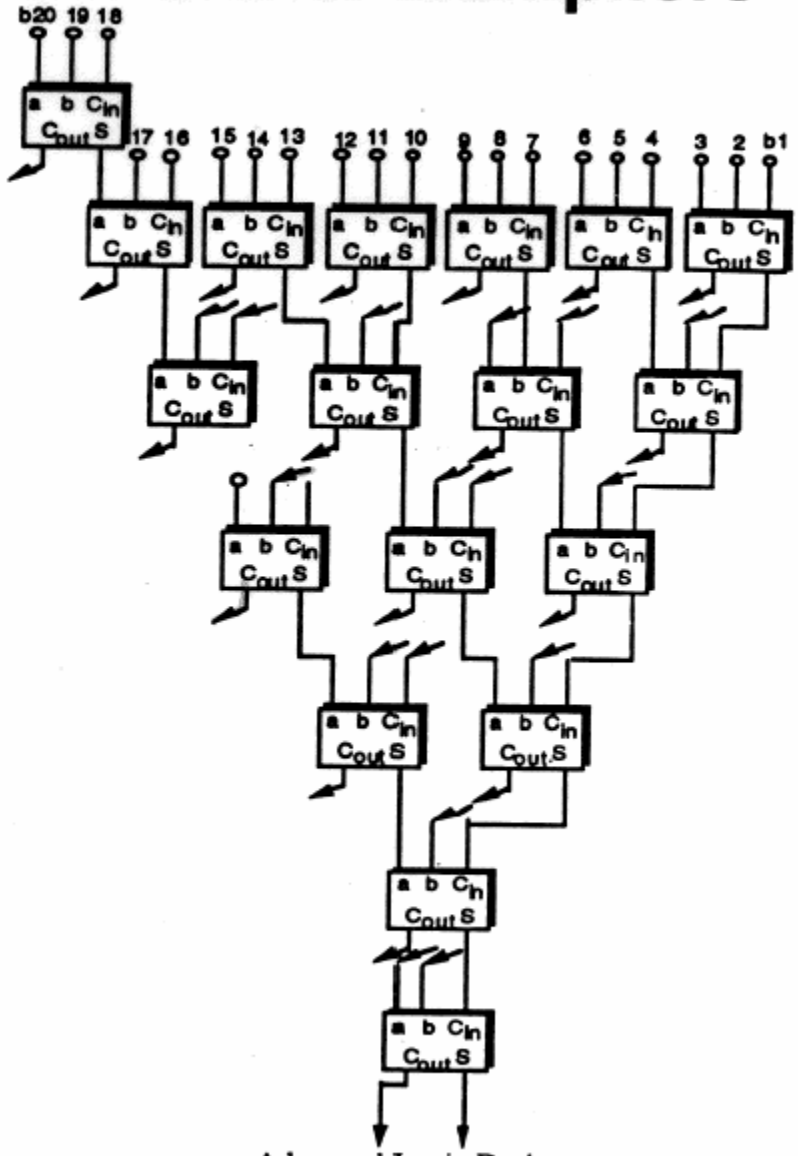


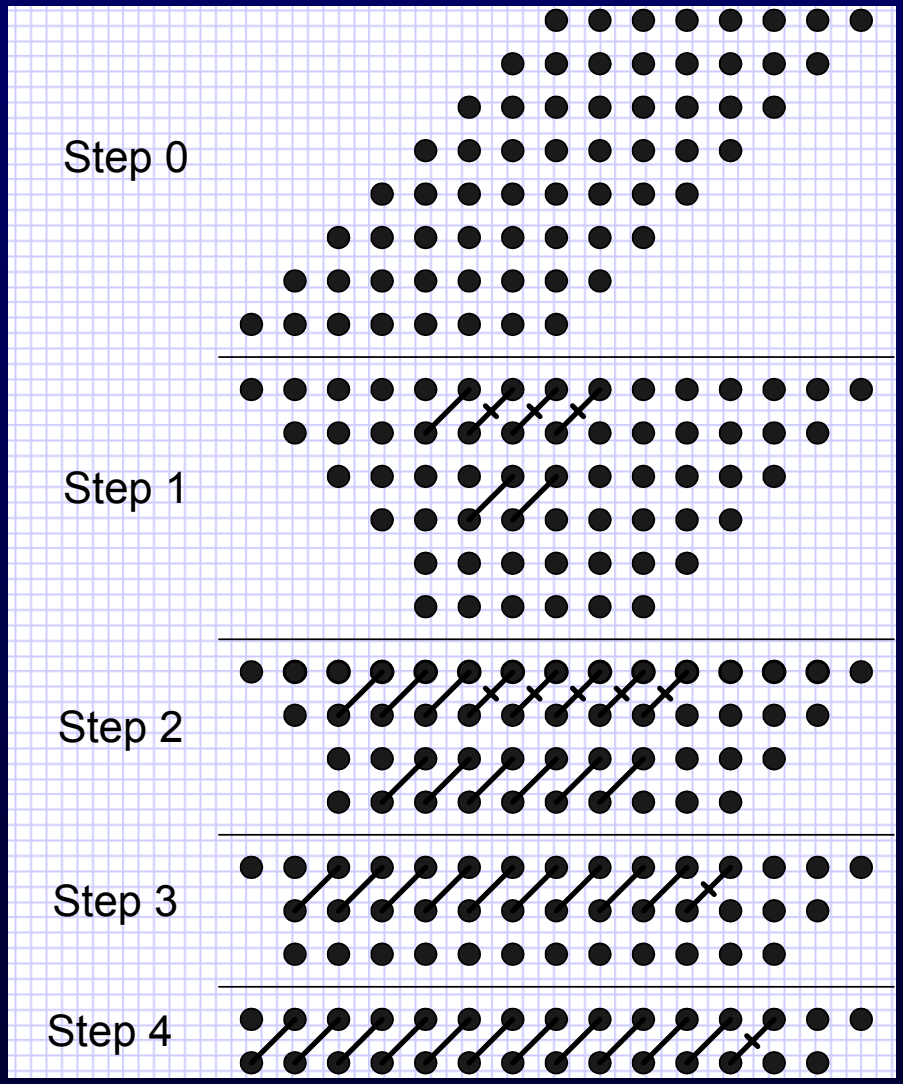
Figure 4–6. Wallace tree

Fast Parallel Multipliers

Wallace:

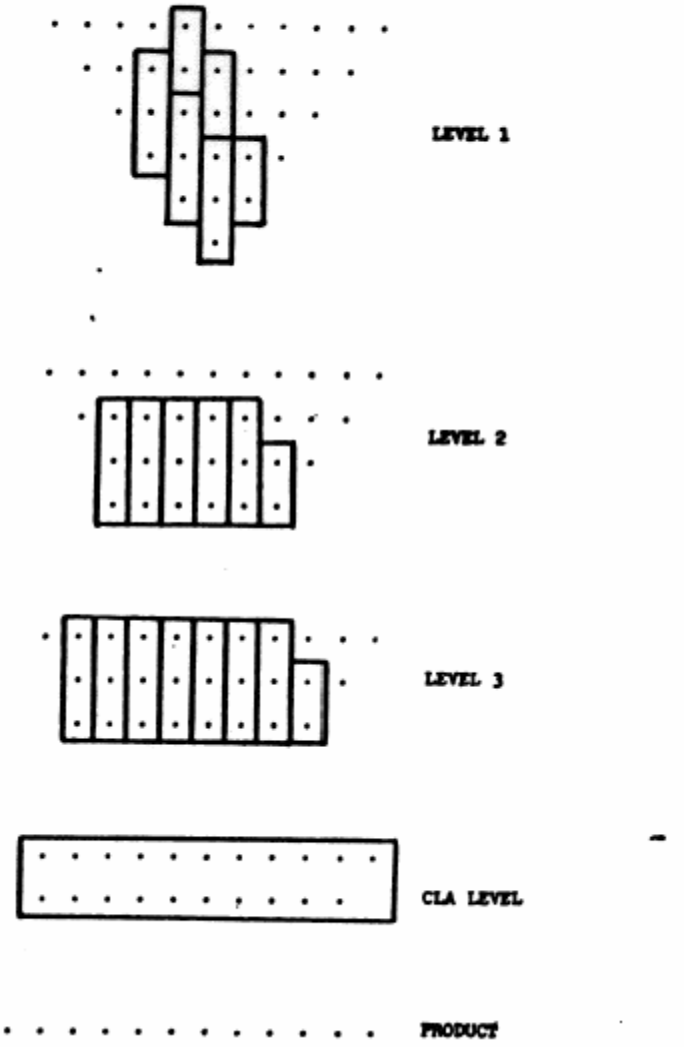


Parallel Multipliers

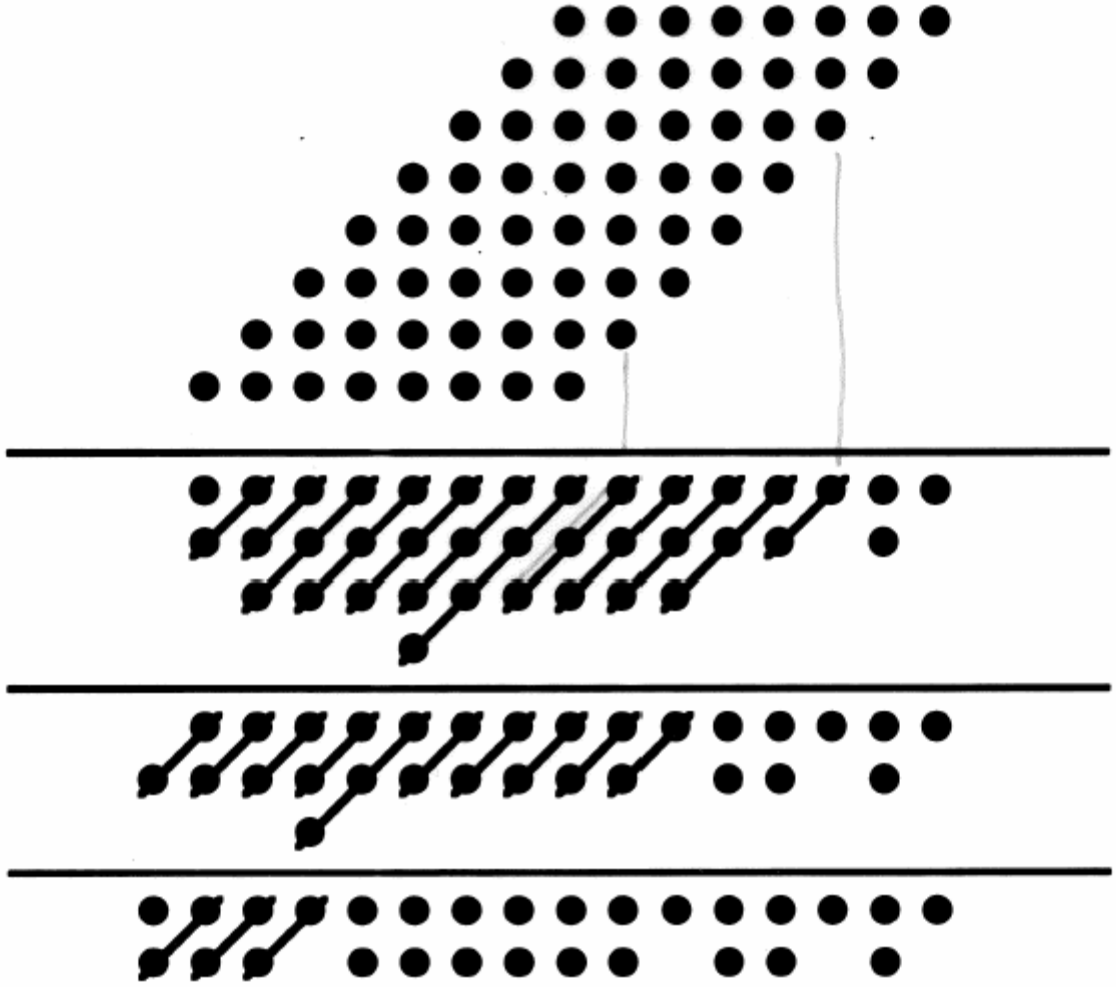


Fast Parallel Multipliers

Wallace:



Bit Reduction Using "Dadda" Counters



Minimum Number of Stages (Dada's Rule)

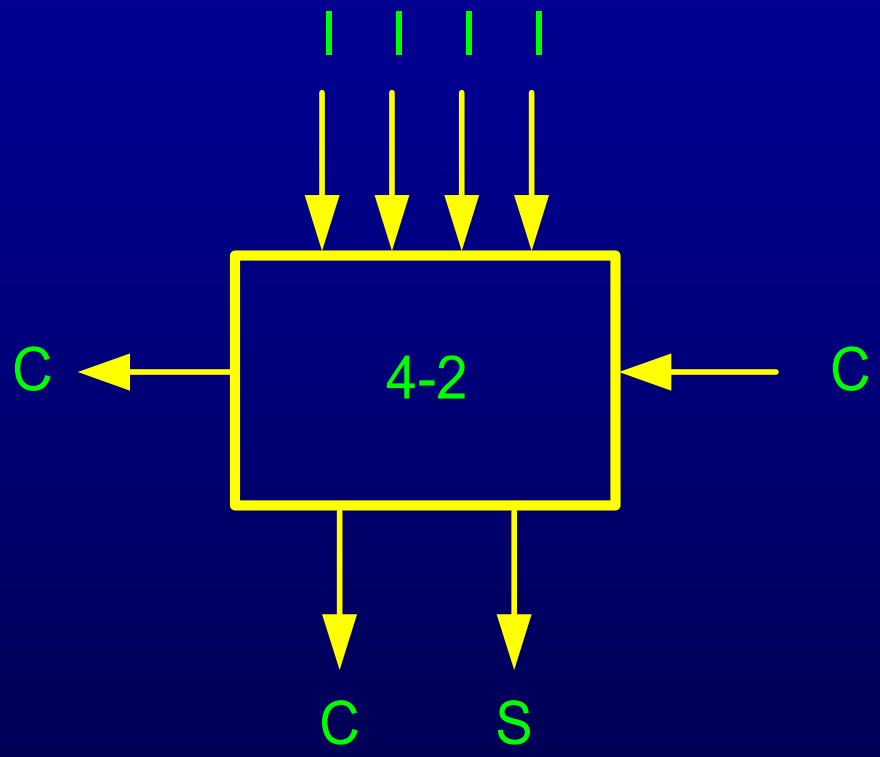
Number of bits in the multiplier	Minimum number of stages
3	1
4	2
$4 < n \leq 6$	3
$6 < n \leq 9$	4
$9 < n \leq 13$	5
$13 < n \leq 19$	6
$19 < n \leq 28$	7
$28 < n \leq 42$	8
$42 < n \leq 63$	9

Use of 4:2 Compressors

A. Weinberger 1981

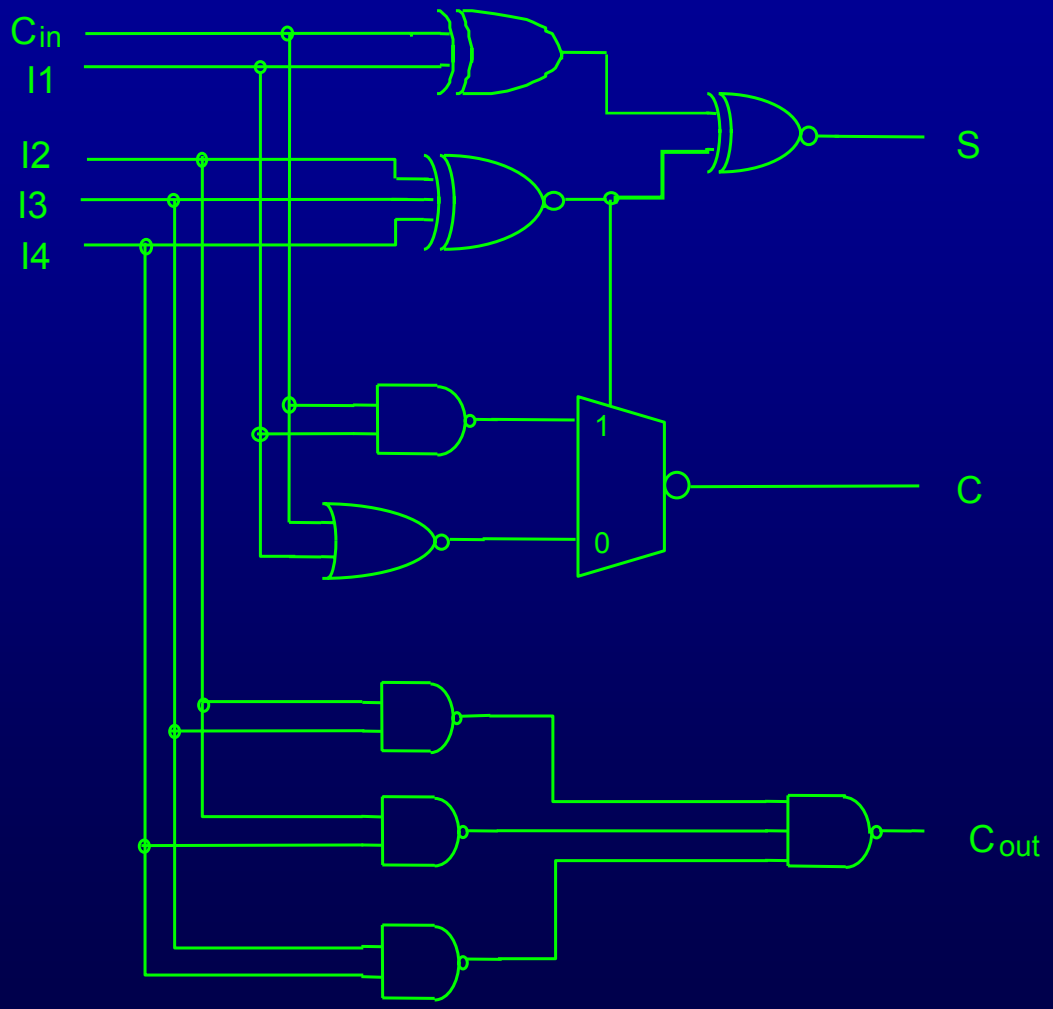
M. Santoro 1988

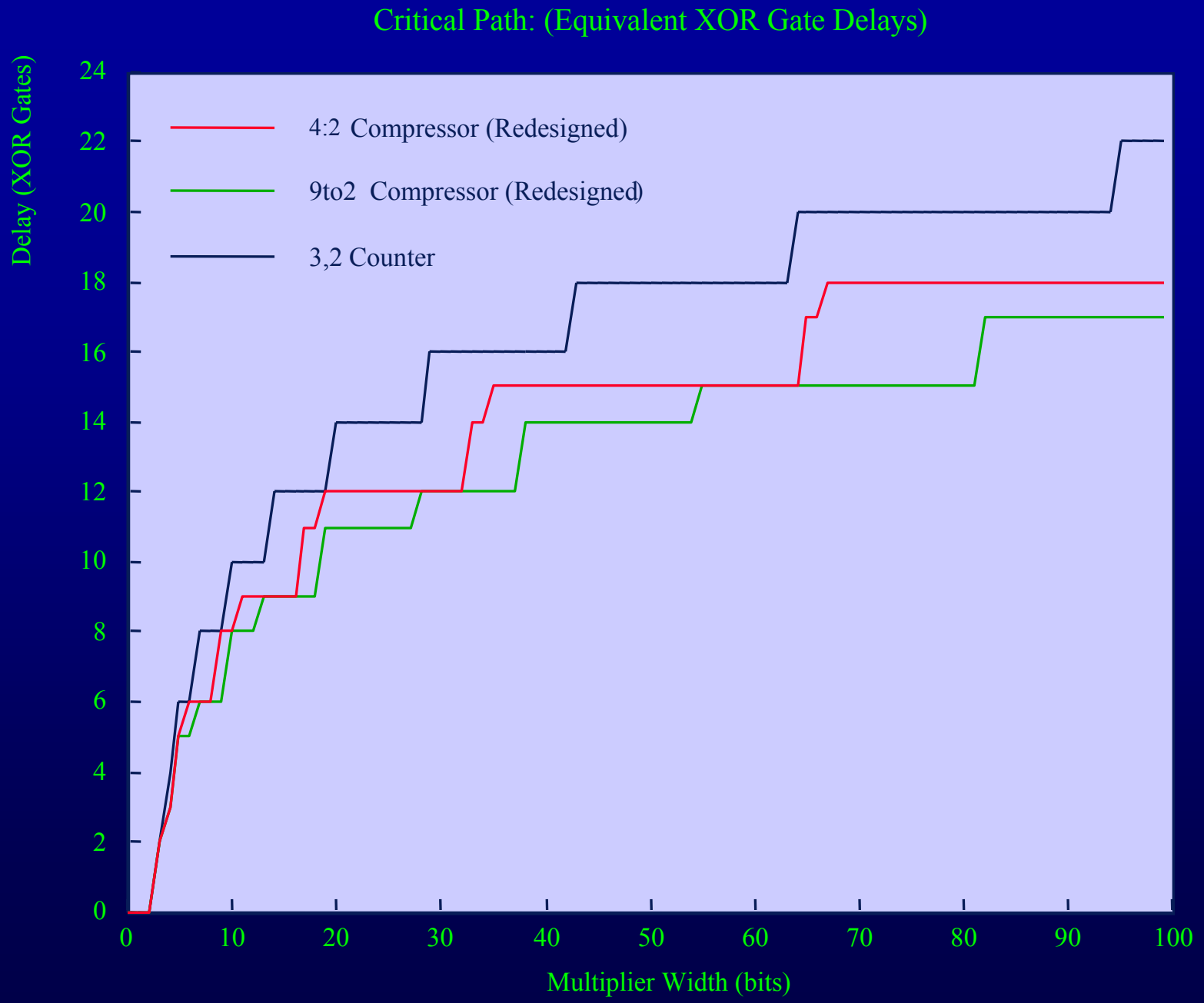
4:2 Compressor



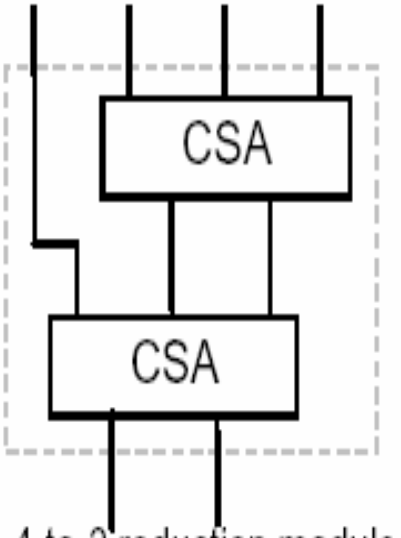
4

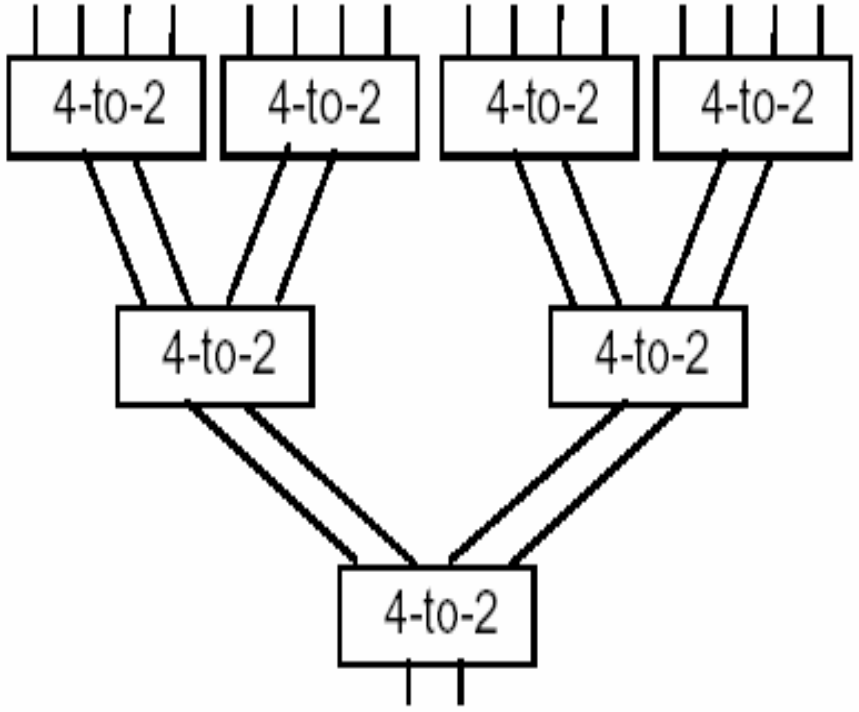
Re-designed 4:2 Compressor with 3 XOR Delay (Nagamatsu, Toshiba)




Critical Path: (Equivalent XOR Gate Delays)
Delay (XOR Gates)
24
22
20
18
16
14
12
10
8
6
4
2
0
4:2 Compressor (Redesigned)
9to2 Compressor (Redesigned)
3,2 Counter
0
10
20
30
40
50
60
70
80
90
100
Multiplier Width (bits)

Tree Multipliers

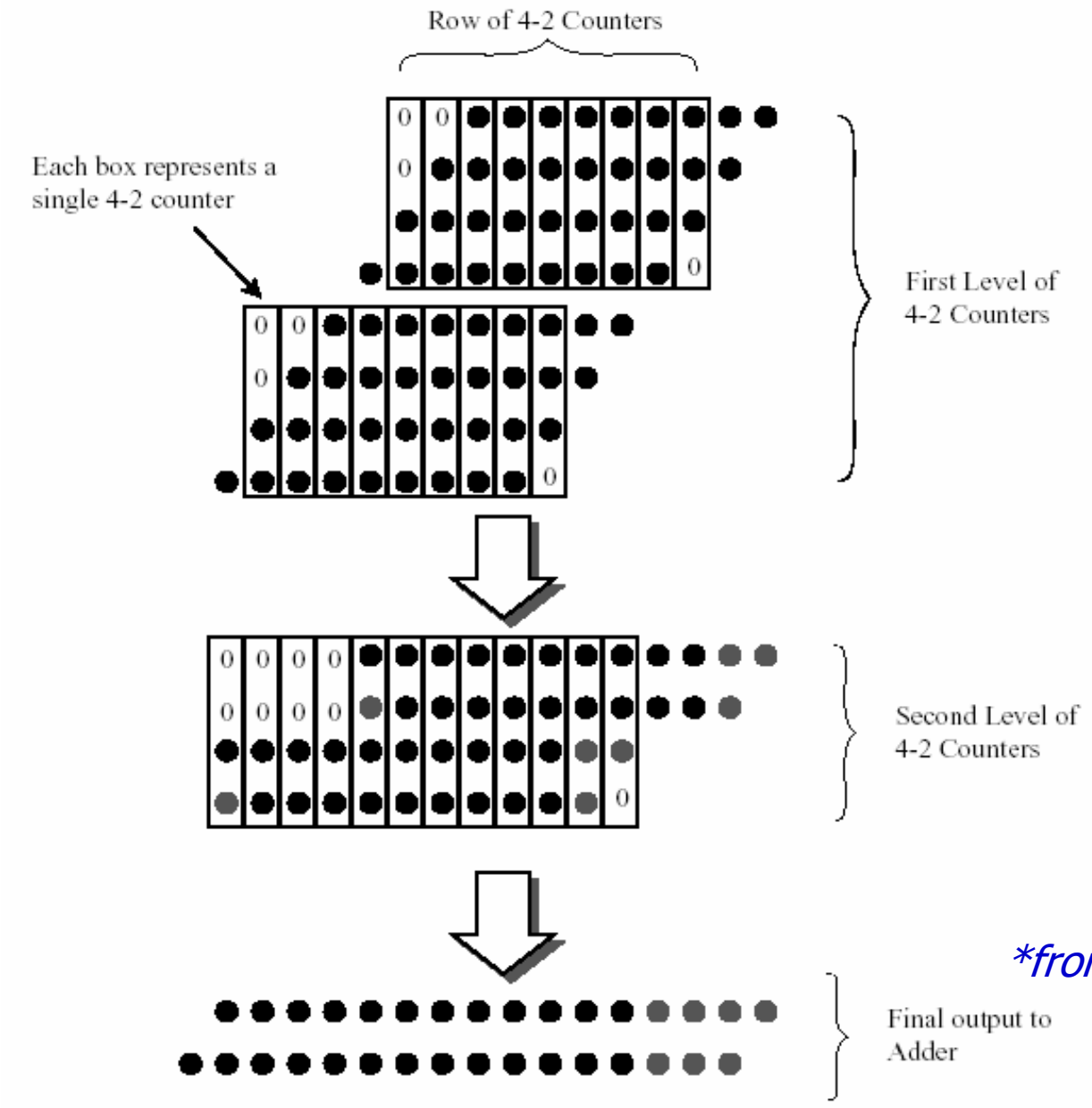




Tree multiplier with a more regular structure based on 4-to-2 reduction modules.

**from Parhami*

Reduction using 4:2 Compressors

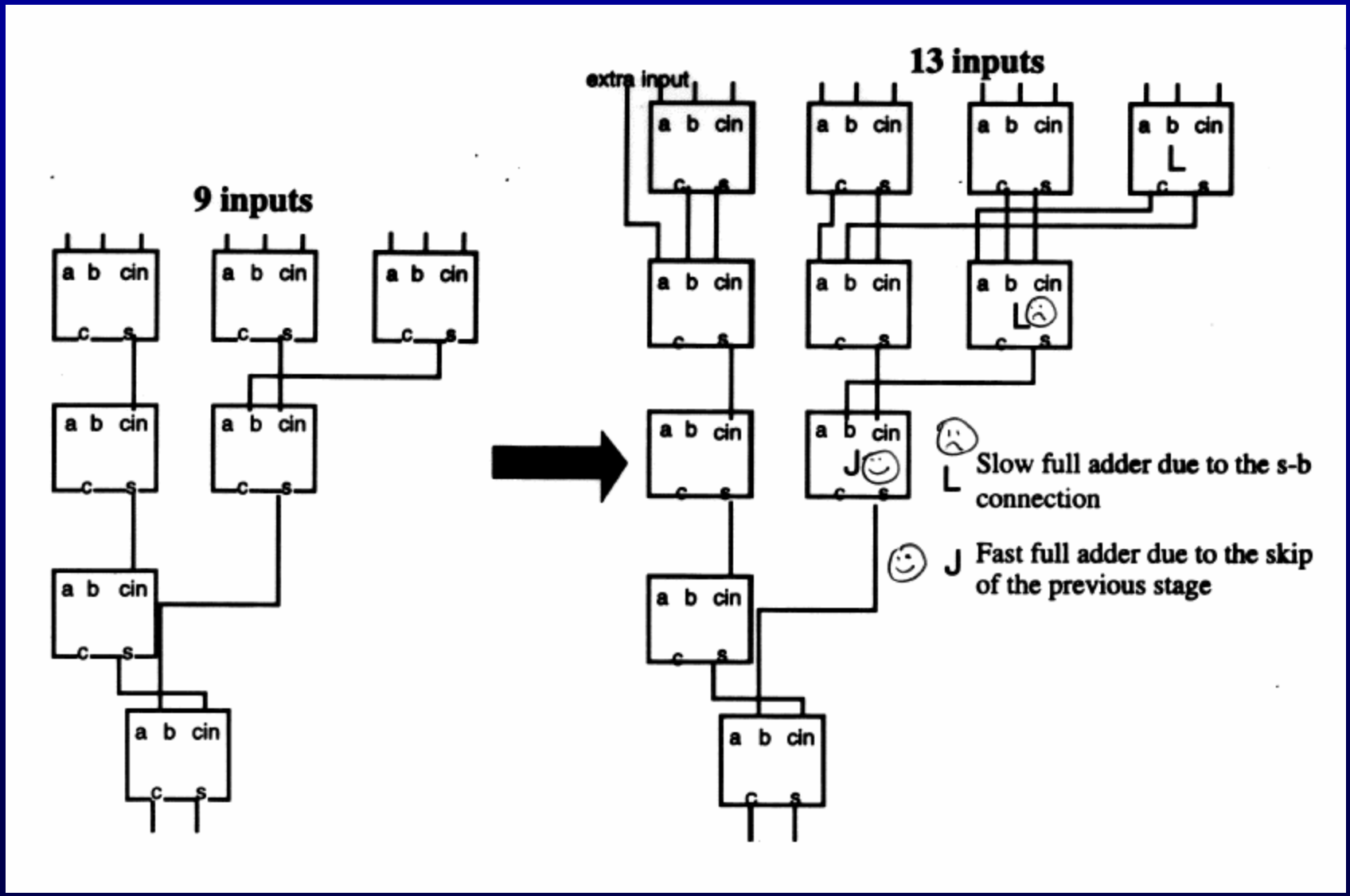


*from G. Bewick

Use of Higher-Order Compressors

D. Villeger, V.G. Oklobdzija 1993

Design of a 13:2 Compressor from a 9:2 Compressor

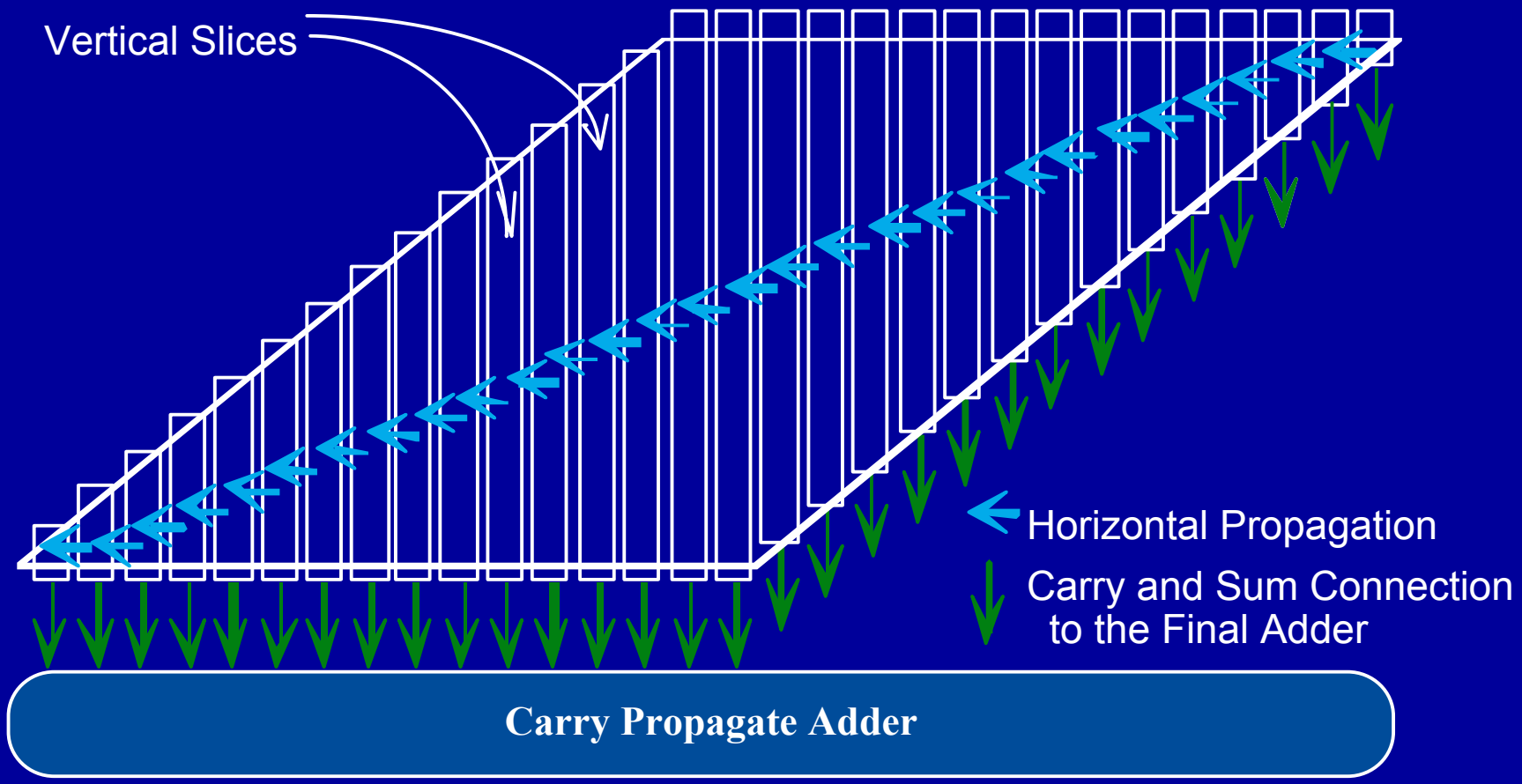


Compressor Family Characteristics

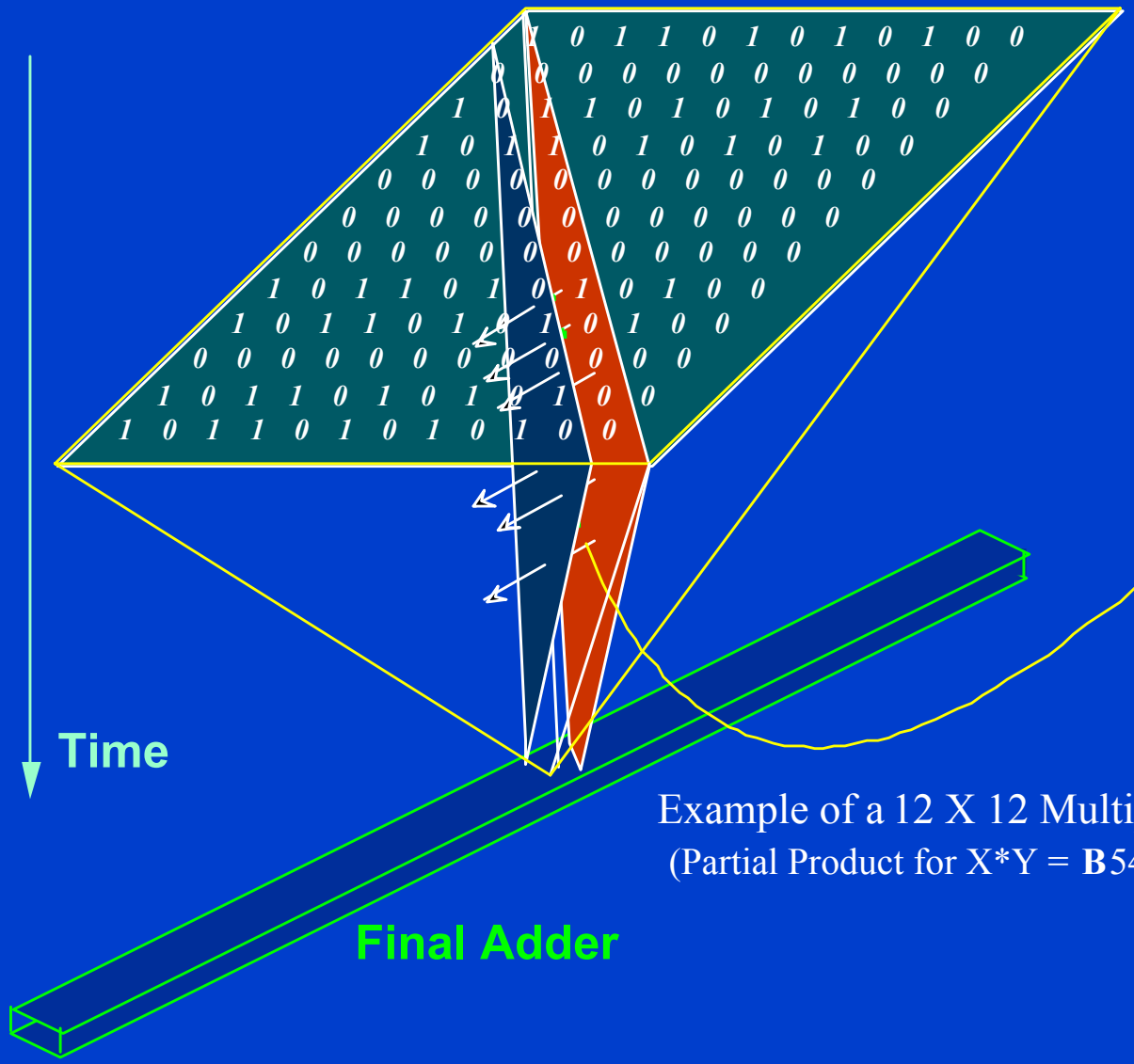
Compressor Counters	No of Full Adder Levels	No XOR Gates
4-2	2	3
6-2	3	5
9-2	4	6
13-2	5	8
18-2	6	9
24-2	7	11
53-2	9	14


Idea !!!!!

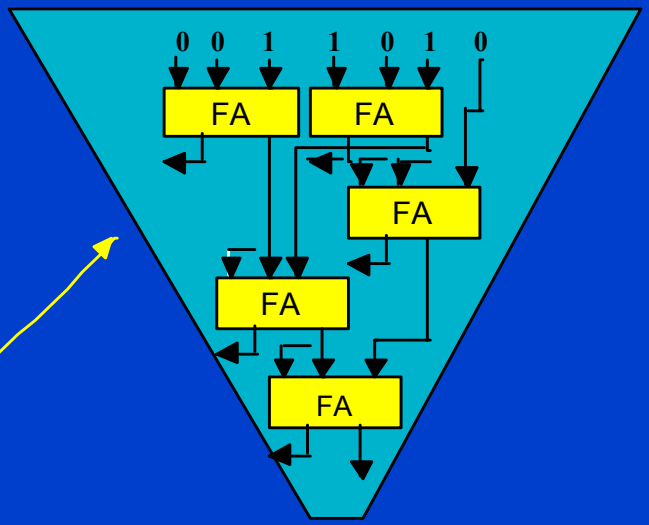
Partial Product Martix Divided into Vertical Compressor Slices



3-Dimensional View of Partial Product Reduction



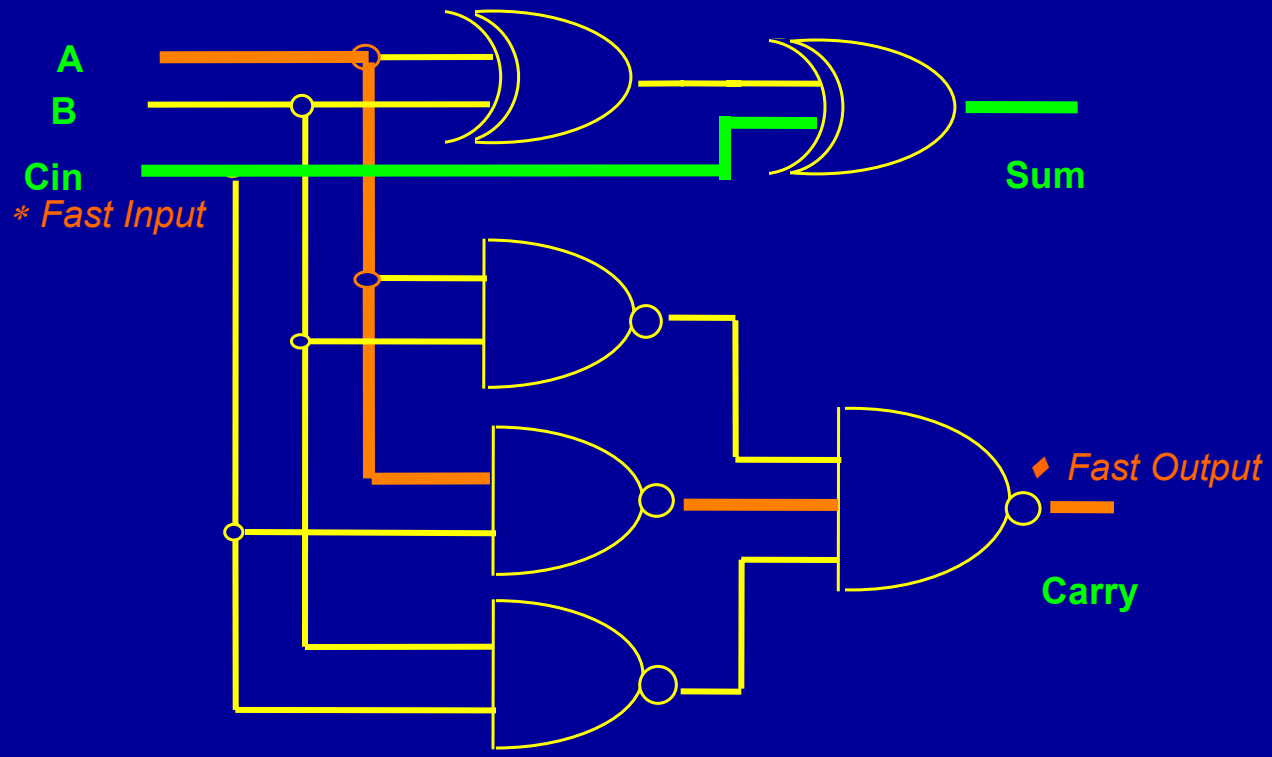
Vertical Compressor Slice - VCS



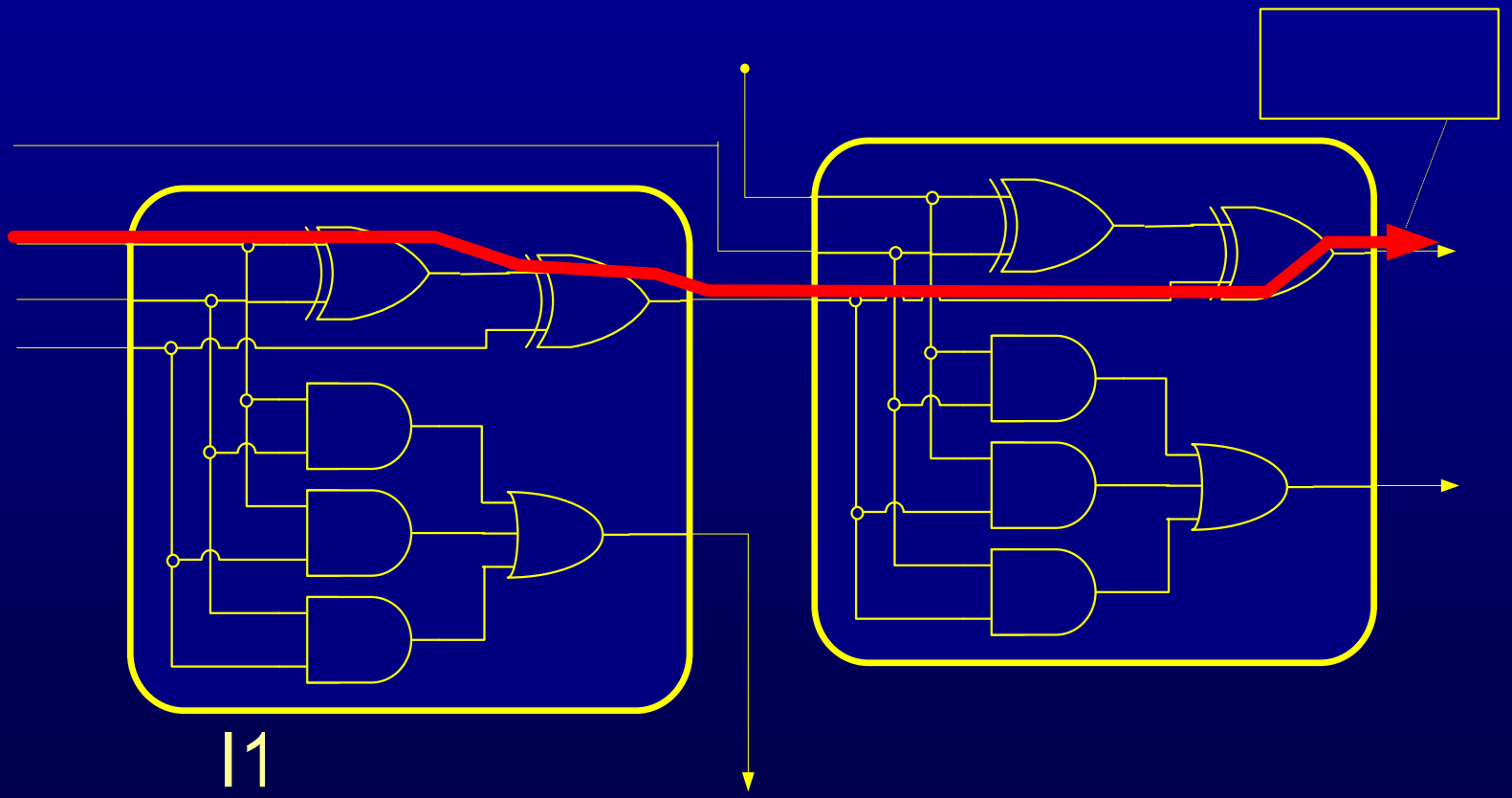
Example of a 12 X 12 Multiplication
(Partial Product for X*Y = **B**54 * **B**1**B**)

Signal Delays in a Full Adder

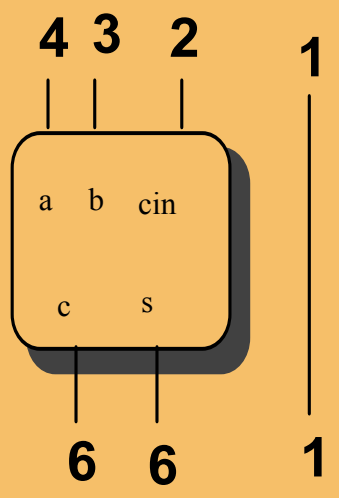
(3,2) Counter



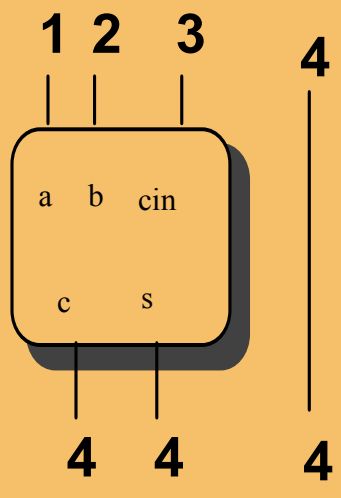
Three-Dimensional optimization Method: TDM (Oklobdzija, Villeger, Liu, 1996)



Method

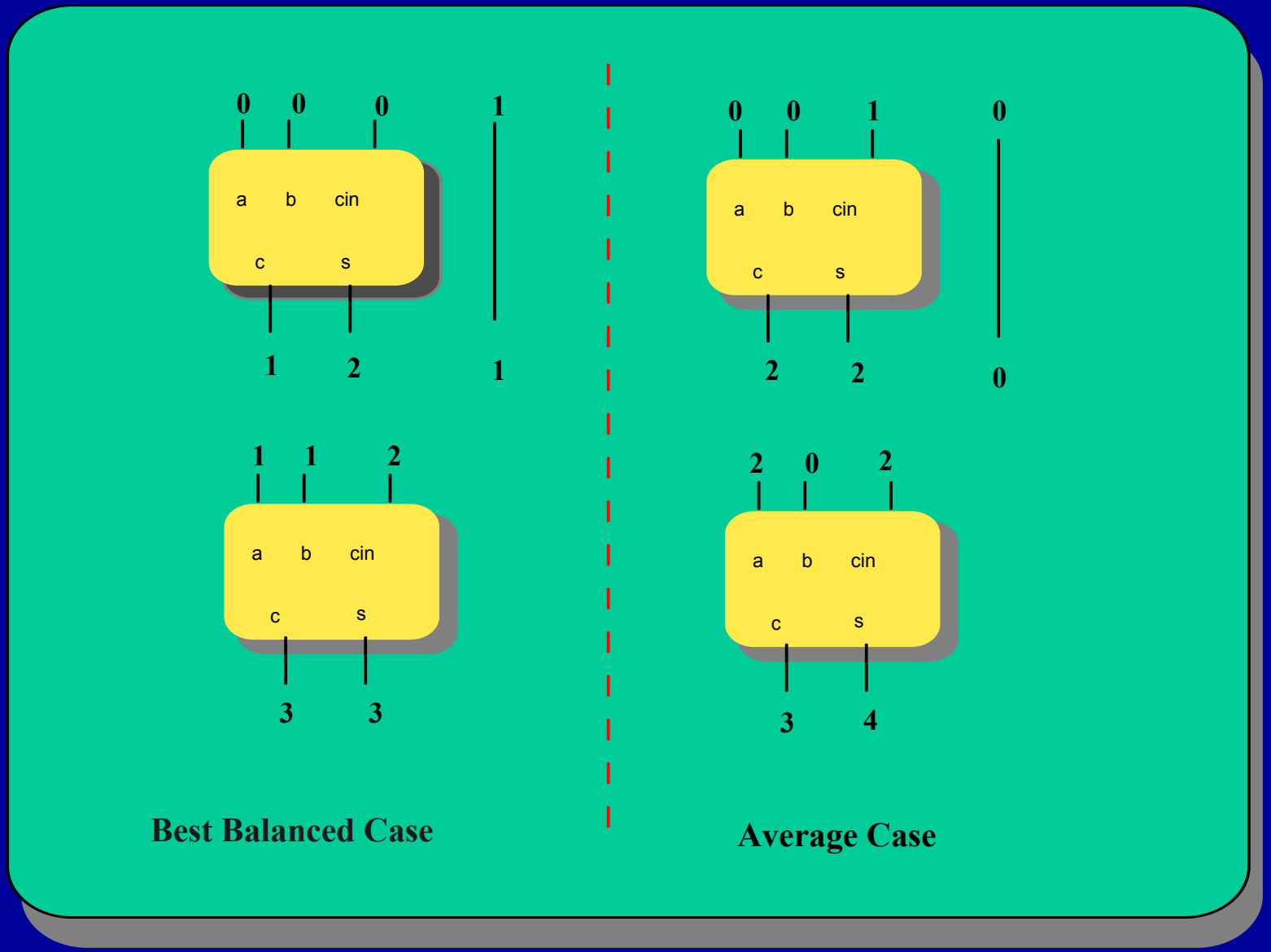


Worst Case



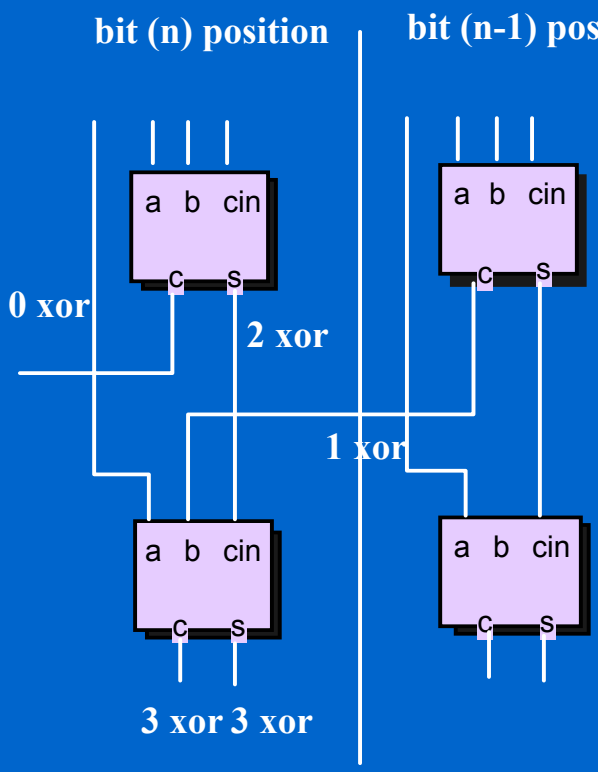
TDM Arrangement

Two cases of signals passing through the next level

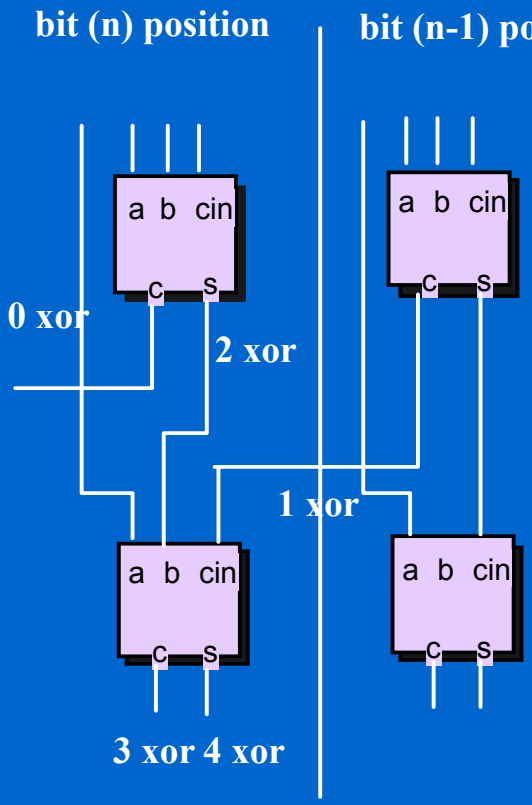


Example of Delay Optimization

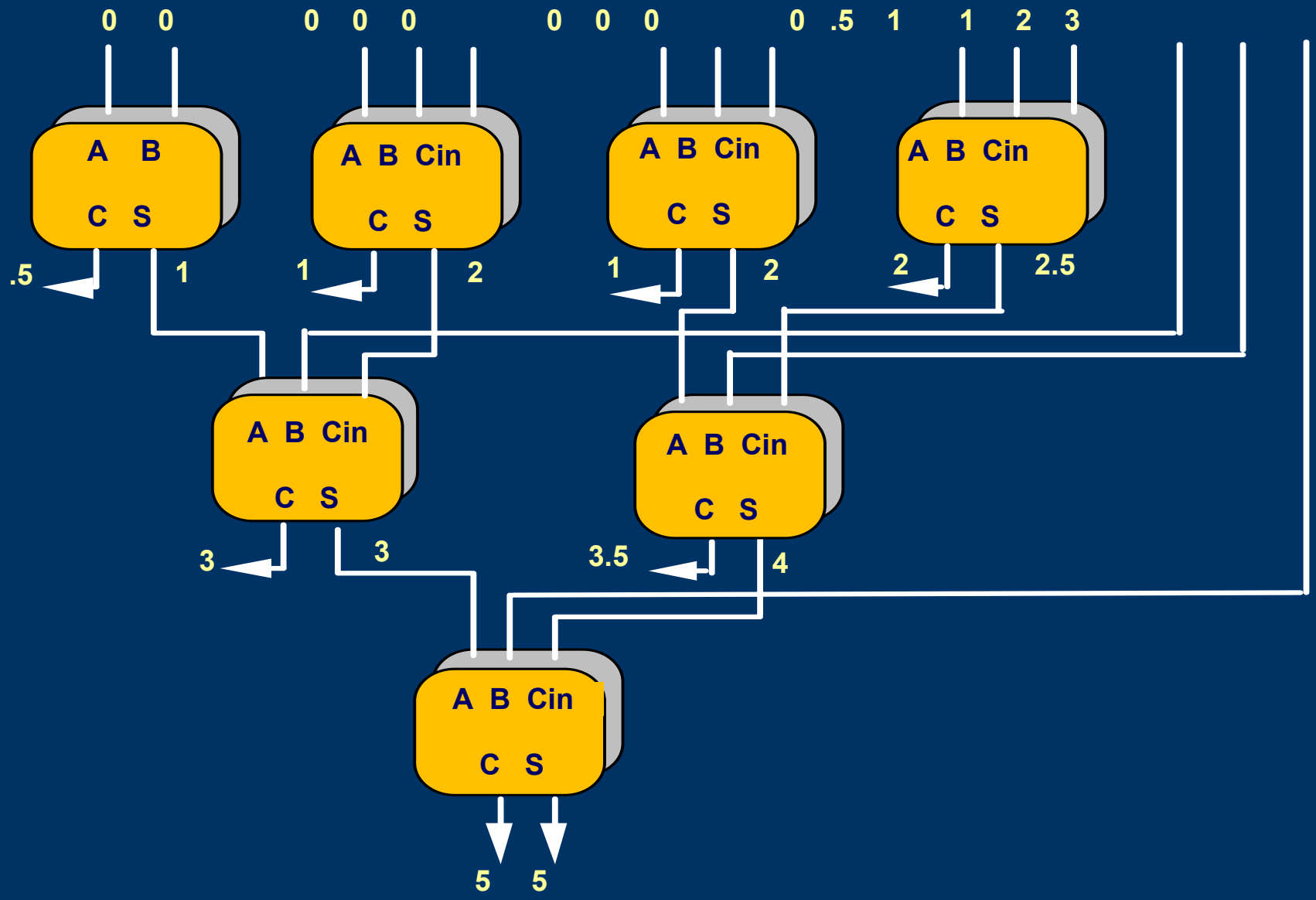
Example of a Optimized Interconnection

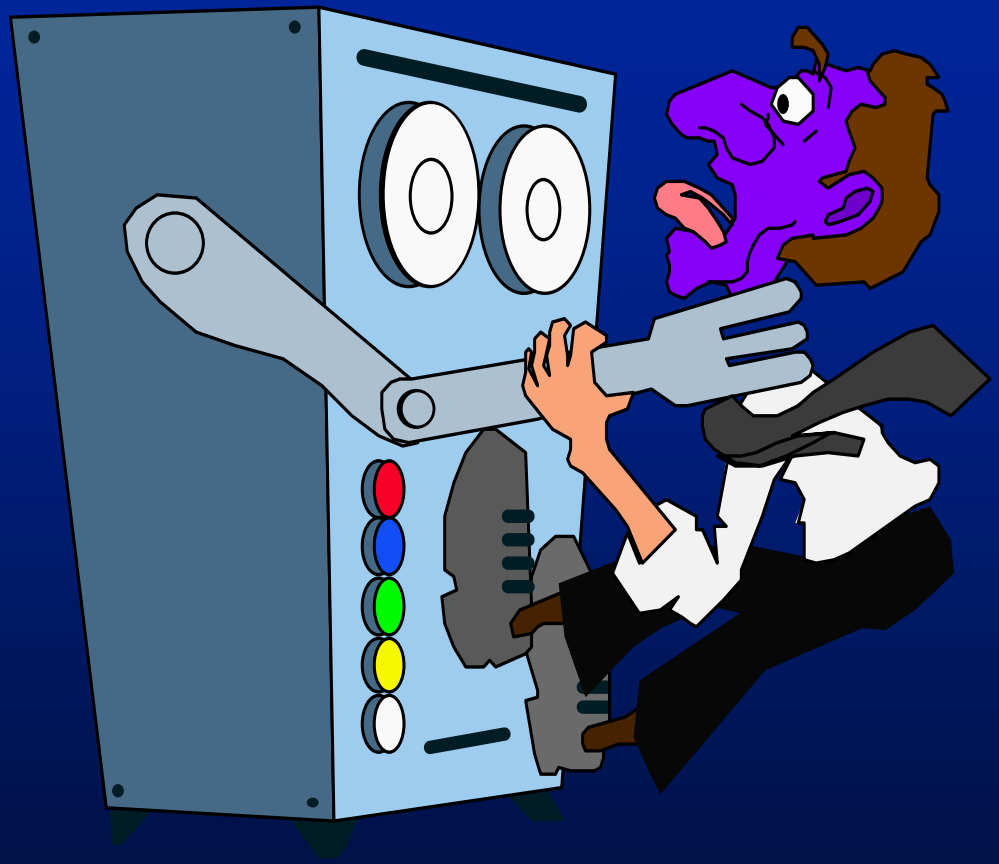


Example of a not Optimized Interconnection



The 9th Vertical Compressor Slice of a Multiplier

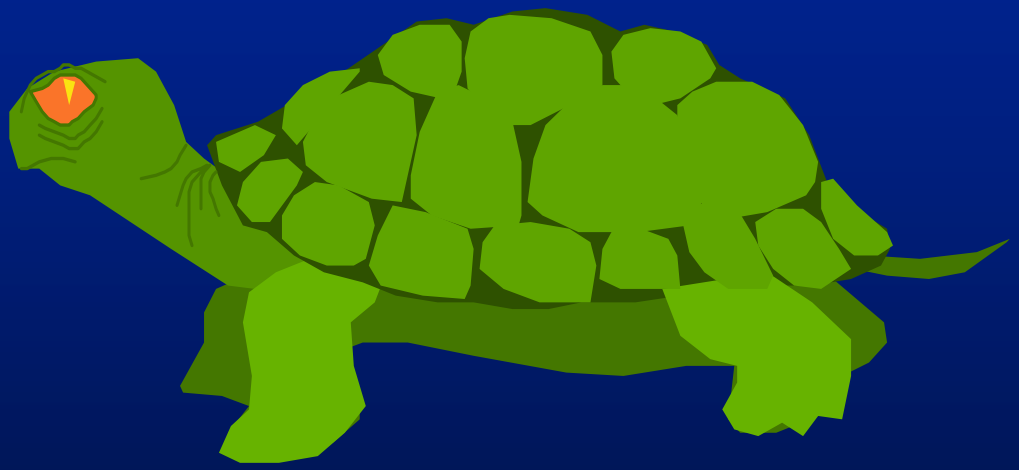


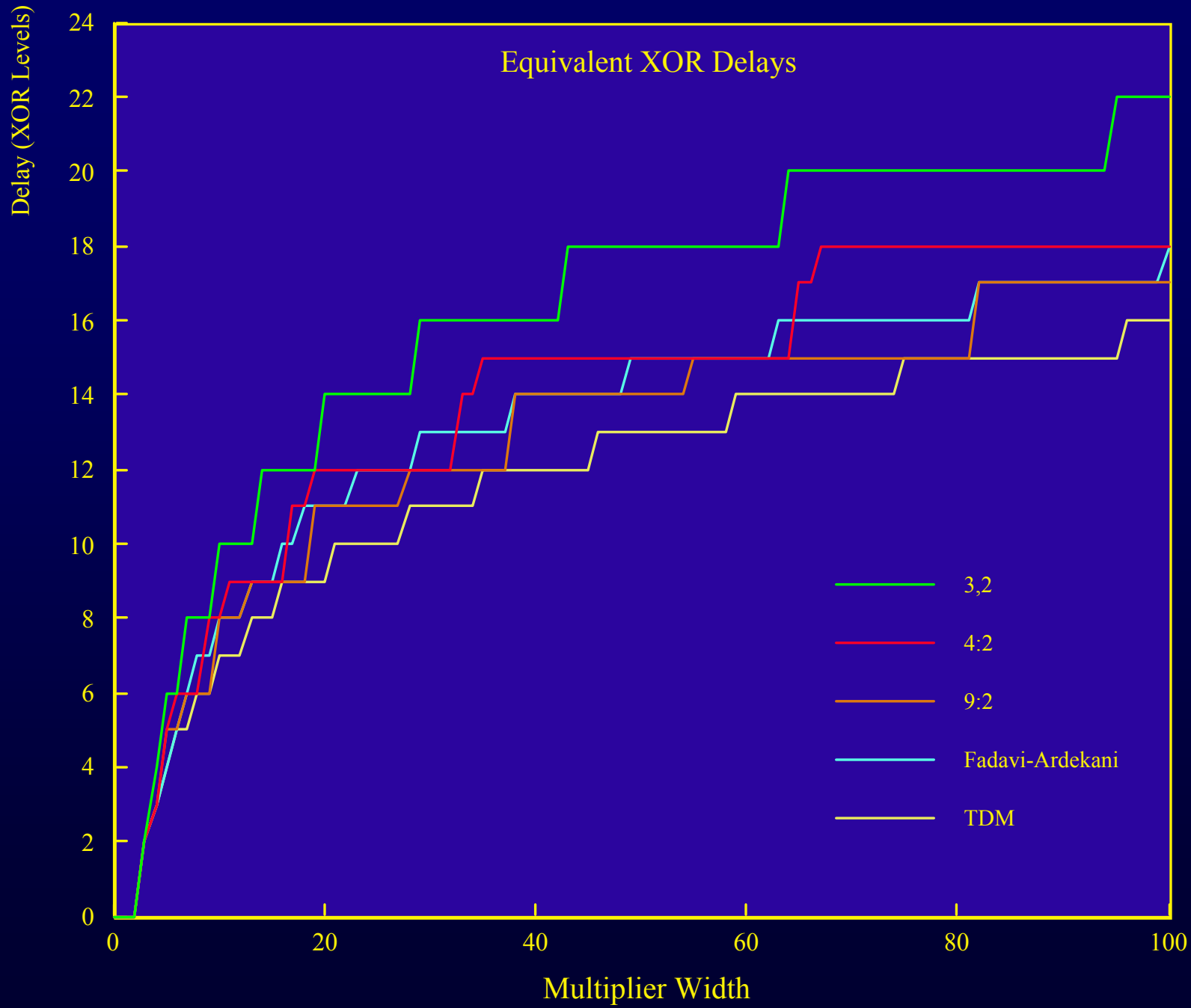


Computer Tools

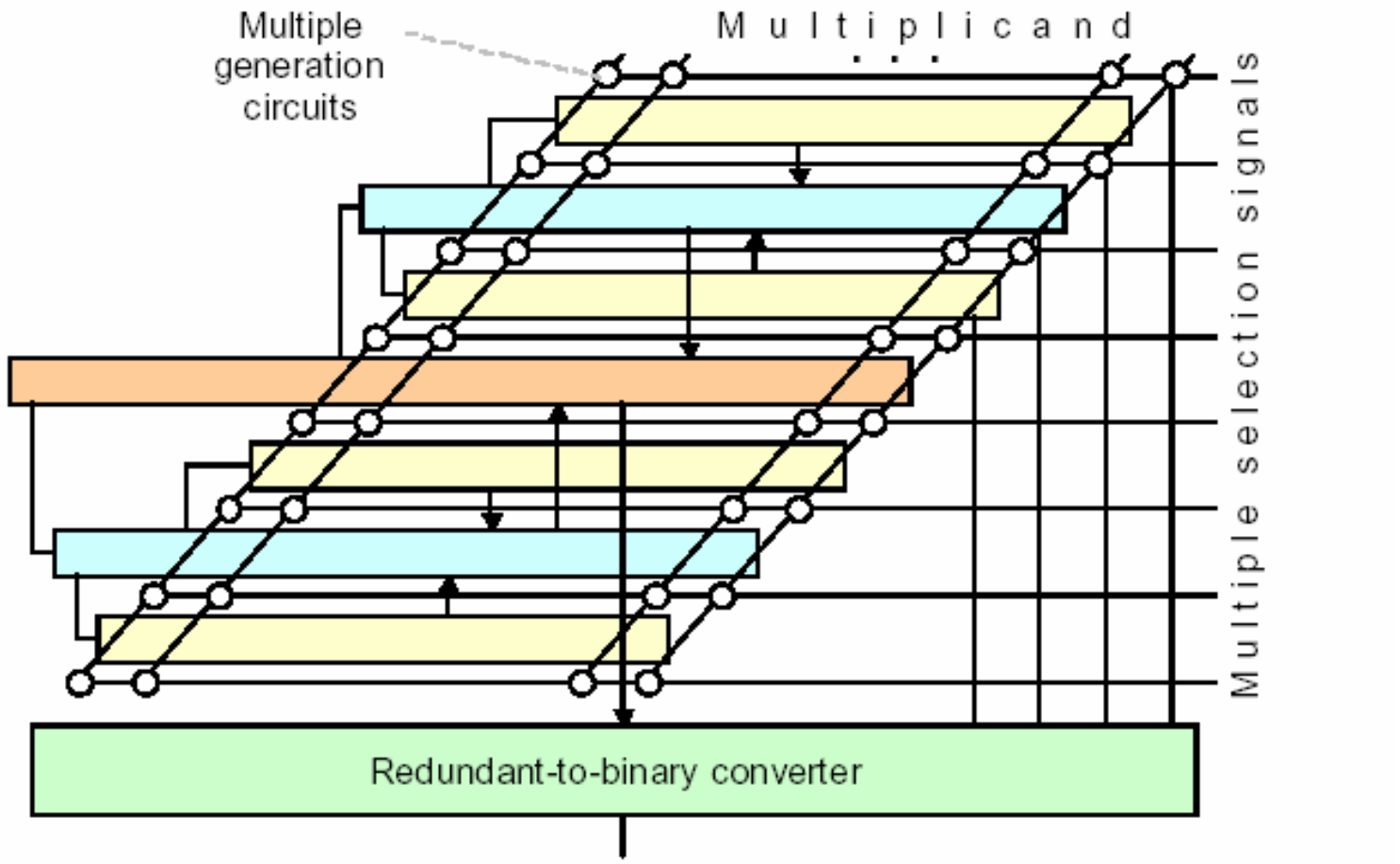
Competing Approaches

Their Schemes




Equivalent XOR Delays
Delay (XOR Levels)
0
2
4
6
8
10
12
14
16
18
20
22
24
0
20
40
60
80
100
Multiplier Width
3,2
4:2
9:2
Fadavi-Ardekani
TDM

Tree Multipliers



Layout of a partial-products reduction tree composed of 4-to-2 reduction modules. Each solid arrow represents two numbers.

**from Parhami*

Floor Plan of a Multiplier

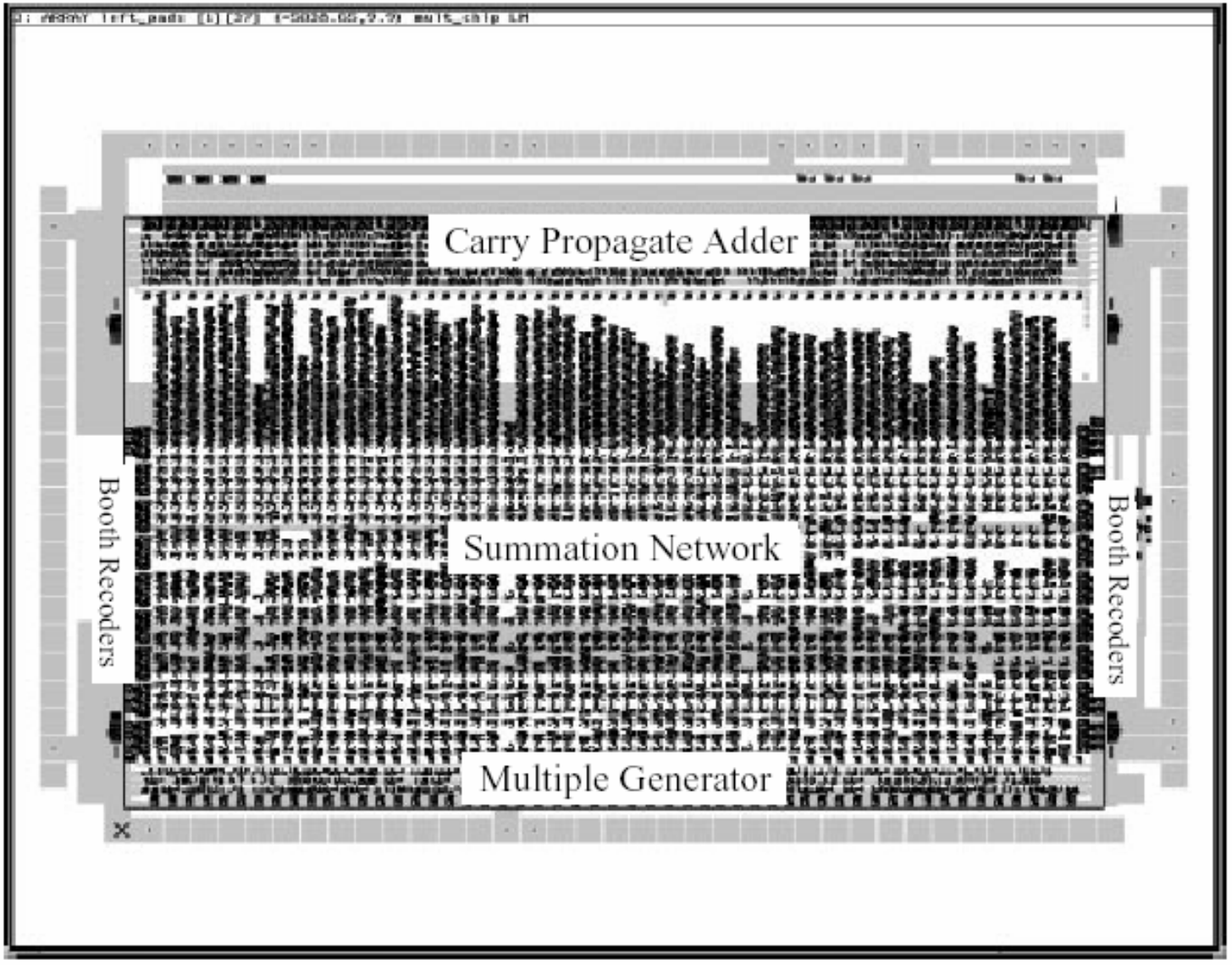
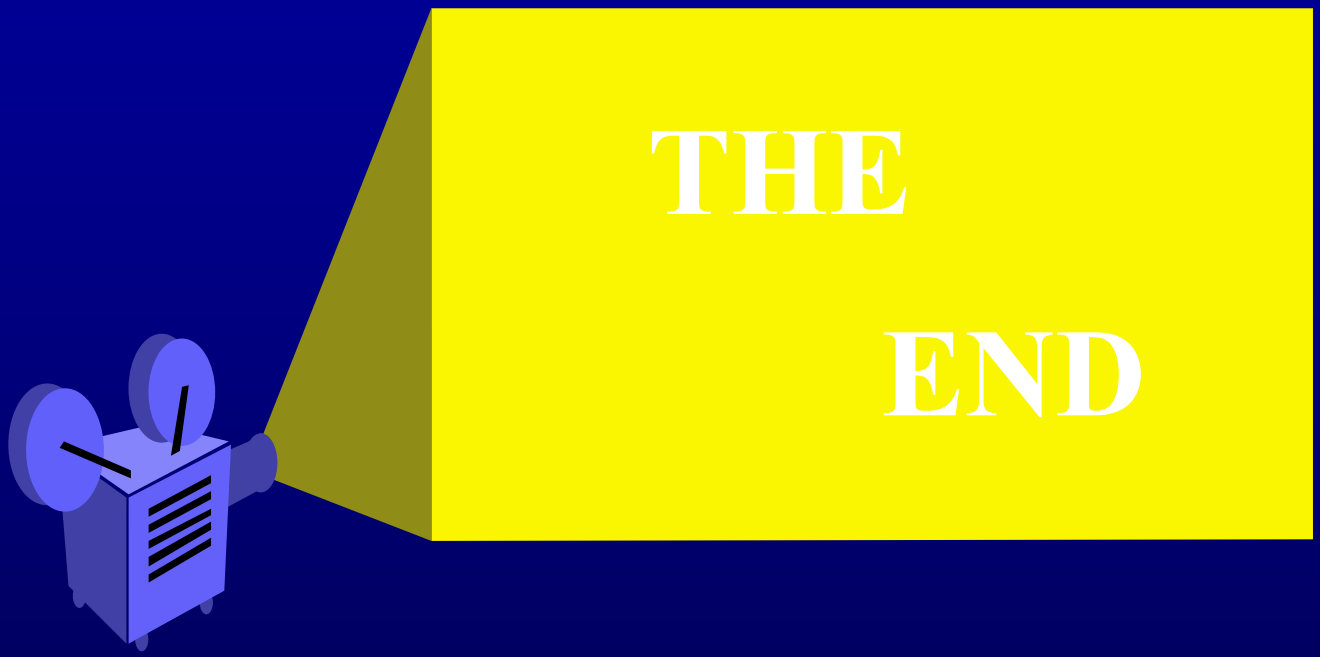
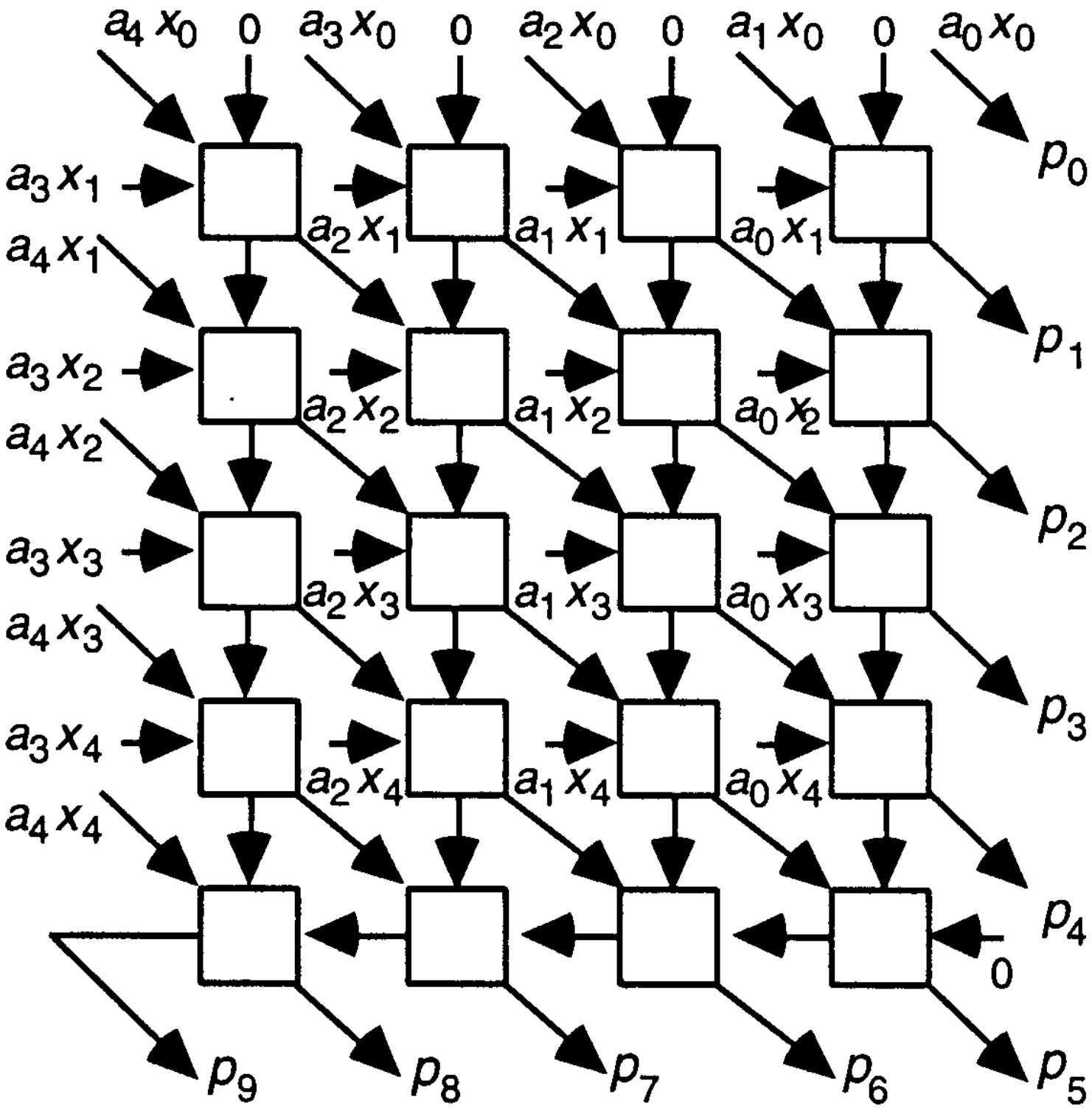


Figure 5.20: Floor plan of multiplier chip

*from G. Bewick


RECOMENDATIONS


THE
END


$a_4 x_0$
0
$a_3 x_0$
0
$a_2 x_0$
0
$a_1 x_0$
0
$a_0 x_0$
p_0
$a_3 x_1$
$a_2 x_1$
$a_1 x_1$
$a_0 x_1$
$a_4 x_1$
p_1
$a_3 x_2$
$a_2 x_2$
$a_1 x_2$
$a_0 x_2$
$a_4 x_2$
p_2
$a_3 x_3$
$a_2 x_3$
$a_1 x_3$
$a_0 x_3$
$a_4 x_3$
p_3
$a_3 x_4$
$a_2 x_4$
$a_1 x_4$
$a_0 x_4$
$a_4 x_4$
p_4
0
p_9
p_8
p_7
p_6
p_5