The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2177-17

ICTP Latin-American Basic Course on FPGA Design for Scientific Instrumentation

15 - 31 March 2010

Digital arithmetic III
(basic arithmetic operations)

BAZARGAN SABET Pirouz

LIP6, University Pierre et Matie Curie
Paris
France

Outline

- Digital CMOS design
- Arithmetic operators
- Sequential functions

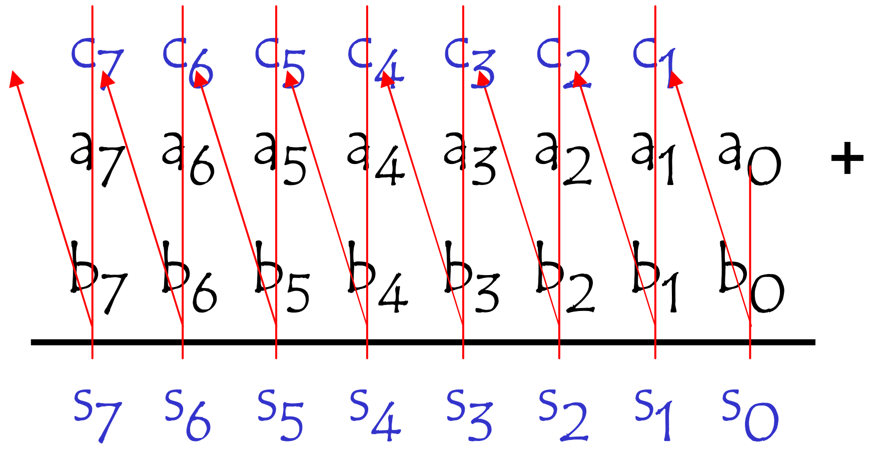
Outline

- Digital CMOS design
- Arithmetic operators
  - Adders
  - Comparators
  - Shifters
  - Multipliers

Adders

Adding two natural numbers

Let consider two natural numbers a and b coded on 8 bits using Natural Binary Code



$c_7 \quad c_6 \quad c_5 \quad c_4 \quad c_3 \quad c_2 \quad c_1$

$a_7 \quad a_6 \quad a_5 \quad a_4 \quad a_3 \quad a_2 \quad a_1 \quad a_0 \quad +$

$b_7 \quad b_6 \quad b_5 \quad b_4 \quad b_3 \quad b_2 \quad b_1 \quad b_0$

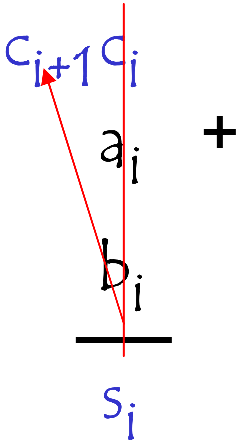
$s_7 \quad s_6 \quad s_5 \quad s_4 \quad s_3 \quad s_2 \quad s_1 \quad s_0$

Adders

Adding two natural numbers

At each stage, I need to sum 3 single bit numbers a_i b_i c_i

The carry out of the stage i is the input carry of the next stage



s_i and c_{i+1} are Boolean functions of a_i b_i c_i

Adders

Adding two natural numbers

c_i \ $a_i b_i$	00	01	11	10
0	0	1	0	1
1	1	0	1	0

s_i

c_i \ $a_i b_i$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

c_{i+1}

$$s_i = a_i \oplus b_i \oplus c_i$$

$$c_{i+1} = a_i.b_i + a_i.c_i + b_i.c_i$$

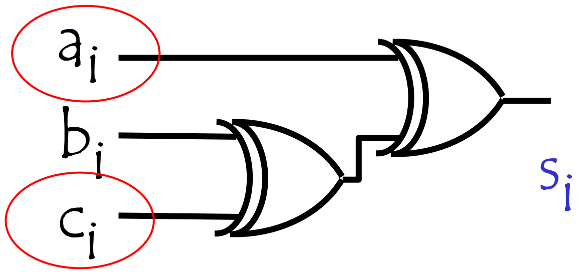
Adders

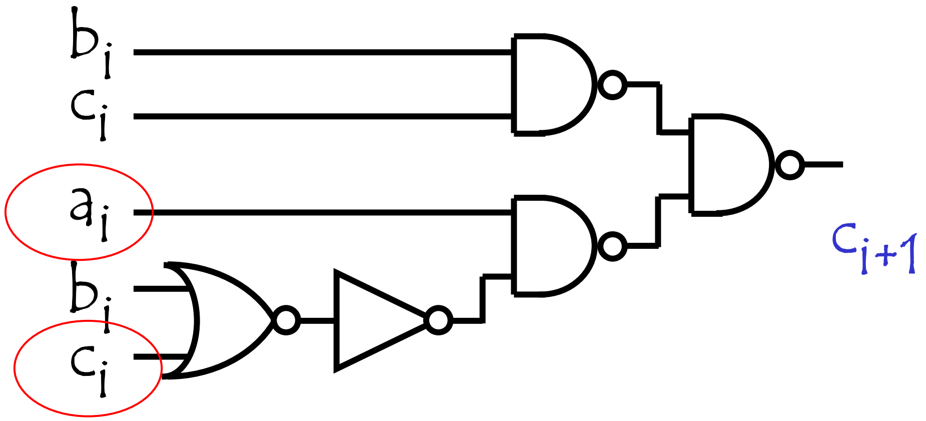
Adding two natural numbers

$$s_i = \underbrace{a_i \oplus \underbrace{b_i \oplus c_i}}$$

$$c_{i+1} = a_i . b_i + a_i . c_i + b_i . c_i$$

$$c_{i+1} = \underbrace{\underbrace{a_i \,.\, \underbrace{(b_i + c_i)}} + \underbrace{b_i . c_i}}$$





Addition delay depends on the delay of c_i to c_{i+1}

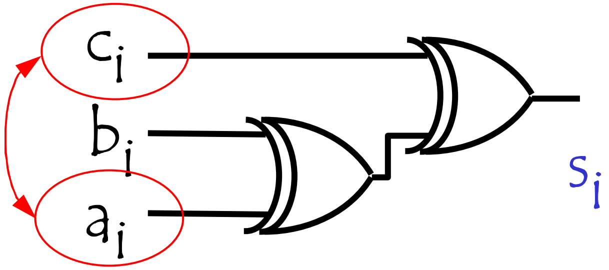
Adders

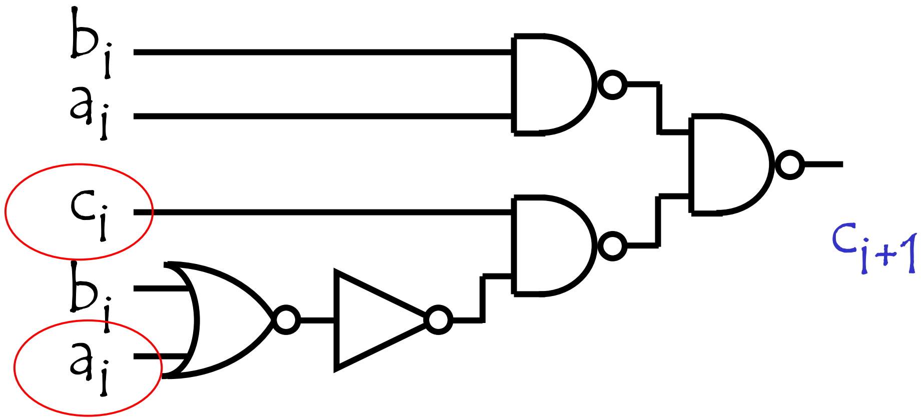
Adding two natural numbers

$s_i = a_i \oplus b_i \oplus c_i$

$c_{i+1} = a_i.b_i + a_i.c_i + b_i.c_i$

$c_{i+1} = a_i.b_i + (a_i+b_i) \cdot c_i$





Adders

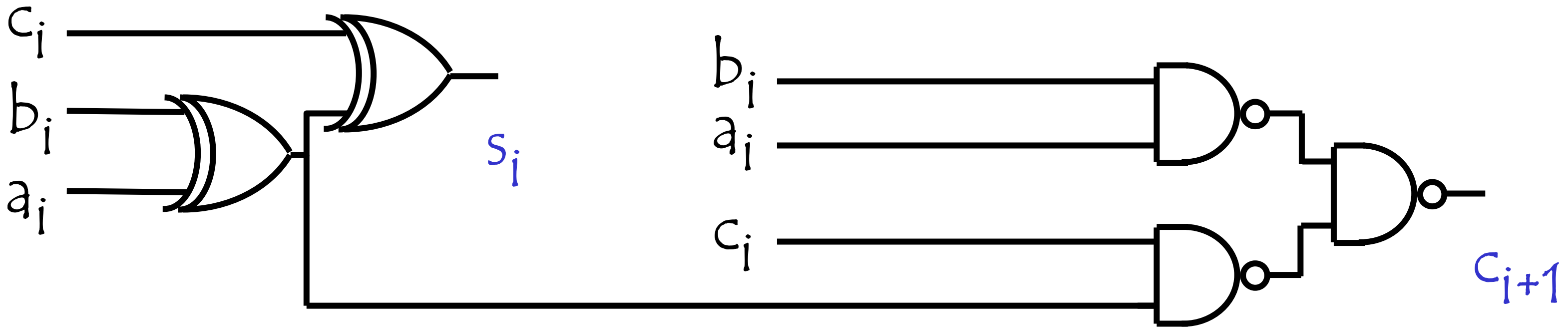
Adding two natural numbers

$s_i = a_i \oplus b_i \oplus c_i$

$c_{i+1} = a_i.b_i + a_i.c_i + b_i.c_i$

$c_{i+1} = a_i.b_i + (a_i+b_i) \,.\, c_i$

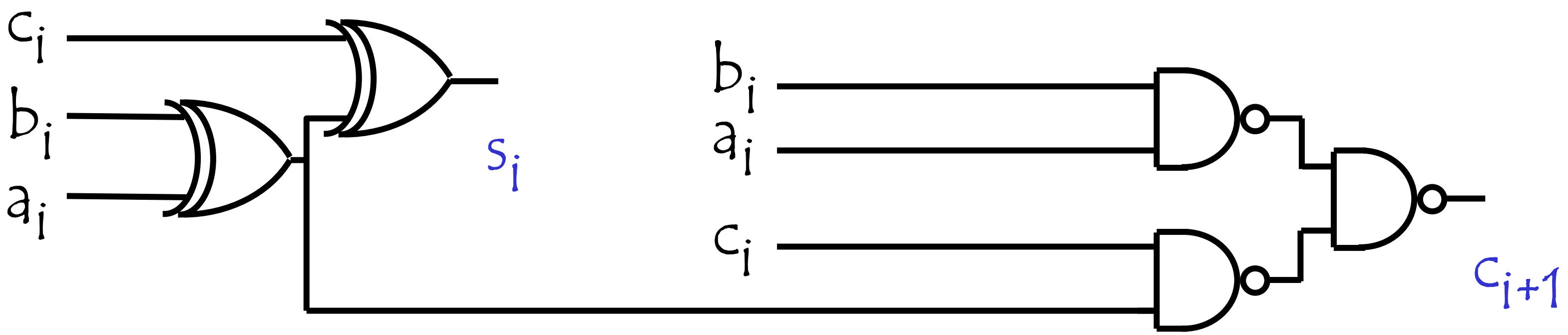
$c_{i+1} = a_i.b_i + (a_i \oplus b_i) \,.\, c_i$



Adders

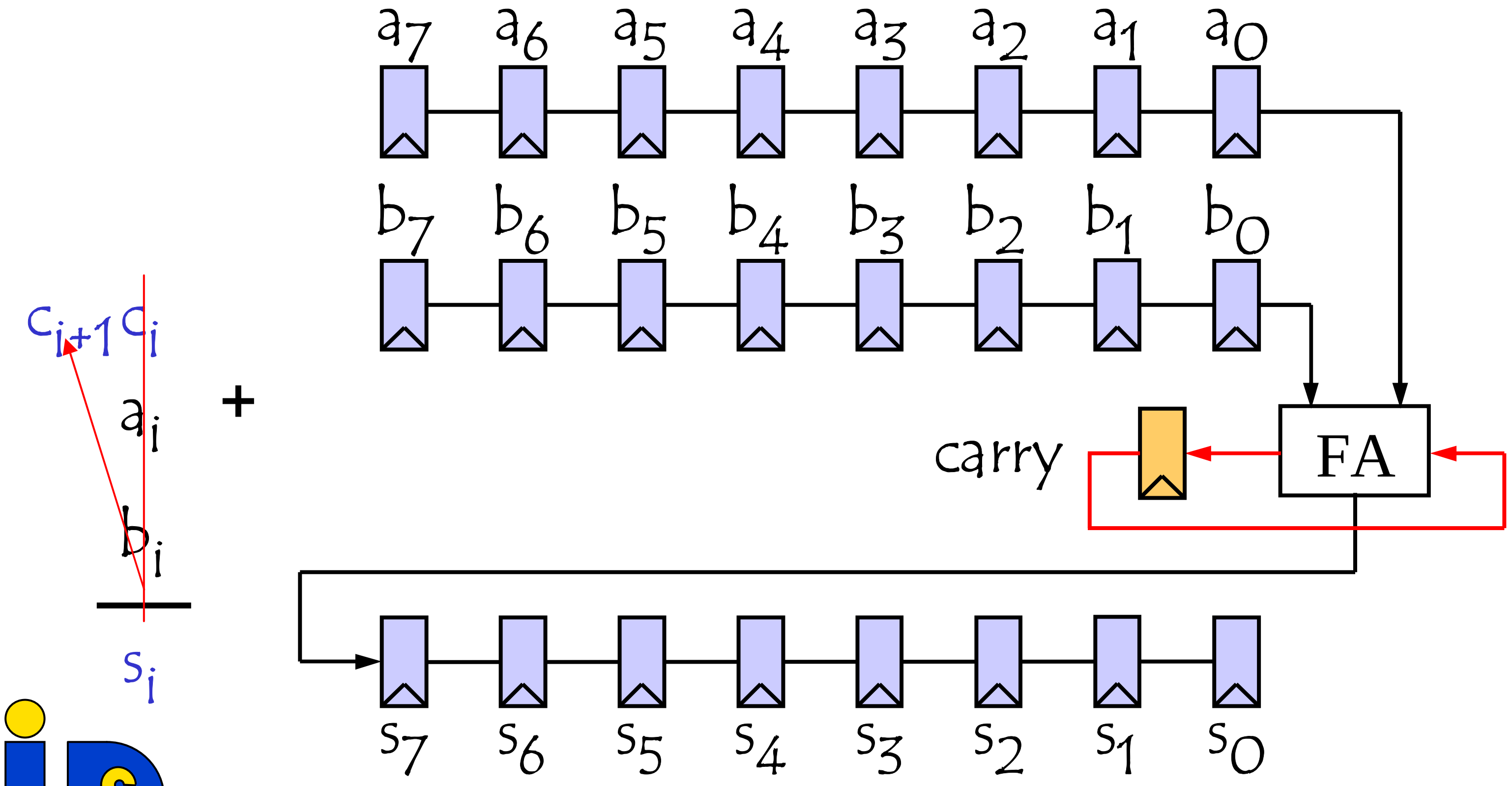
Adding two natural numbers

The circuit generating s_i and c_{i+1} is called a Full Adder (FA)



Adders

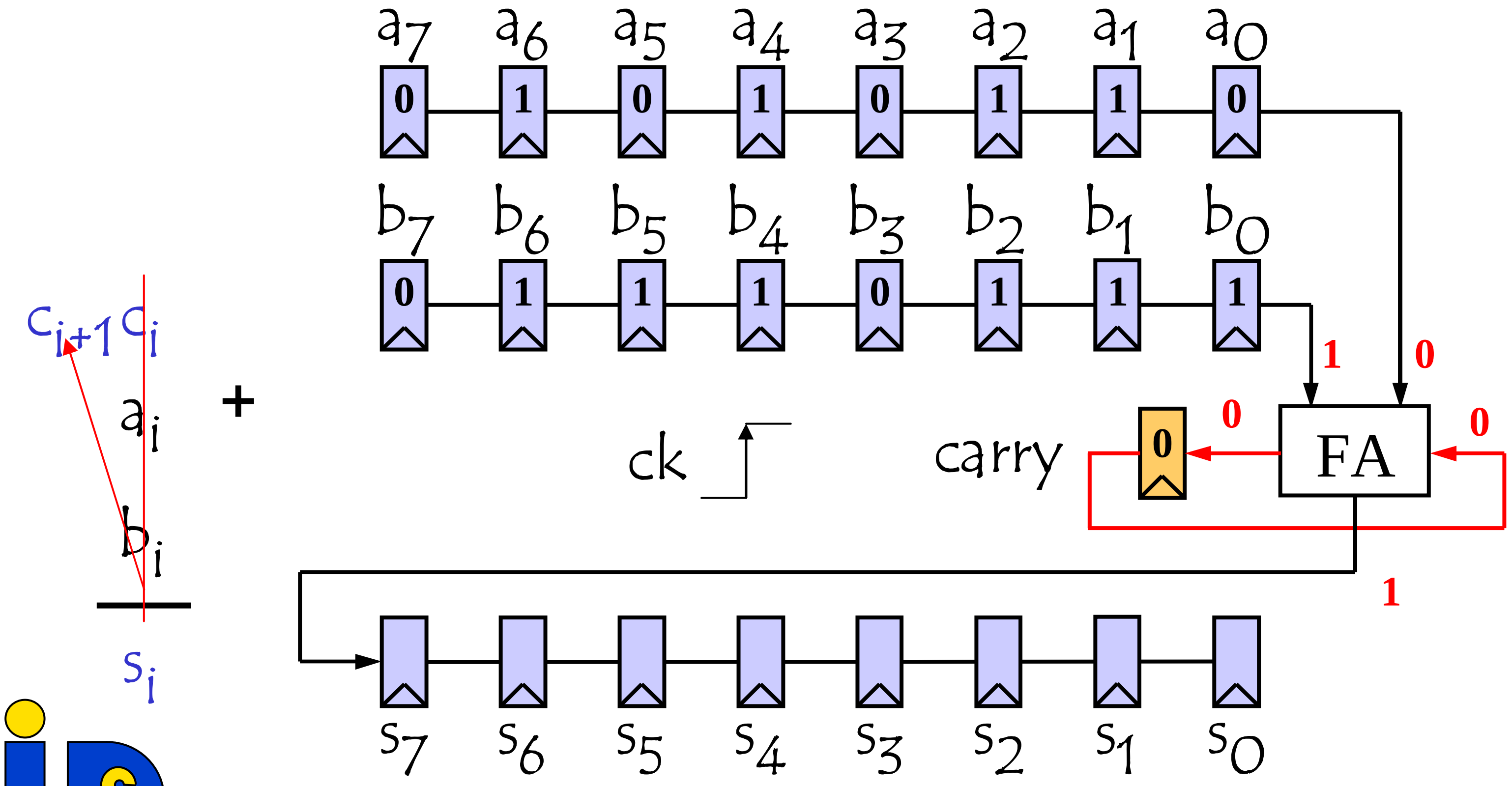
Adding two natural numbers



c_{i+1} c_i
a_i
b_i
s_i
+
a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0
carry
FA
s_7 s_6 s_5 s_4 s_3 s_2 s_1 s_0

Adders

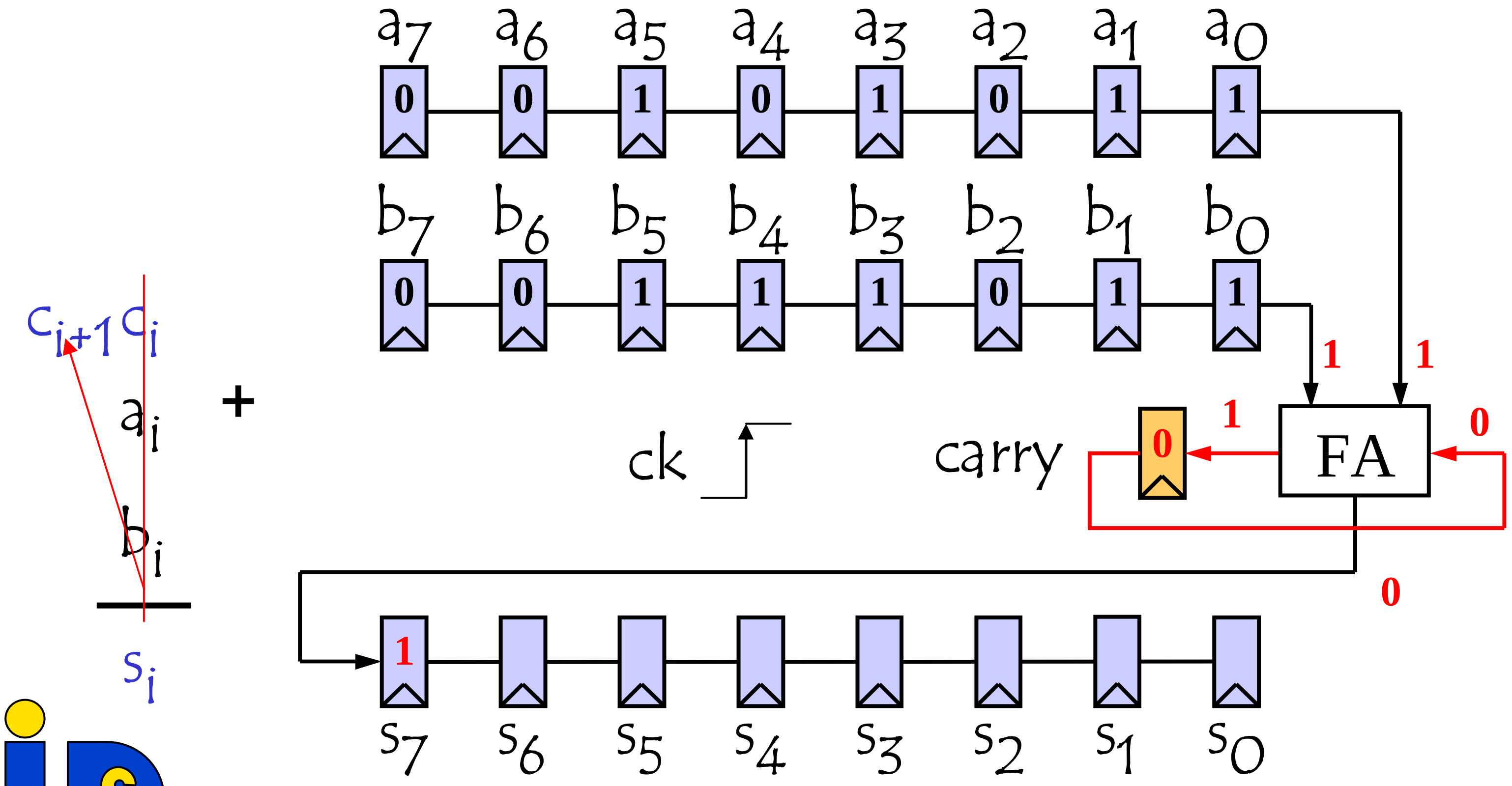
Adding two natural numbers



c_{i+1} c_i
a_i
b_i
s_i
+
a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
0 1 0 1 0 1 1 0
b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0
0 1 1 1 0 1 1 1
1
0
ck
carry
0
0
FA
0
1
s_7 s_6 s_5 s_4 s_3 s_2 s_1 s_0

Adders

Adding two natural numbers


a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0
0 0 1 0 1 0 1 1
b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0
0 0 1 1 1 0 1 1
c_{i+1} c_i
a_i
b_i
s_i
+
ck
carry
0
1
FA
1
1
0
0
1
s_7 s_6 s_5 s_4 s_3 s_2 s_1 s_0

Adders

Adding two natural numbers

Sequential Adder

Area ∝ n

Delay ∝ n cycles

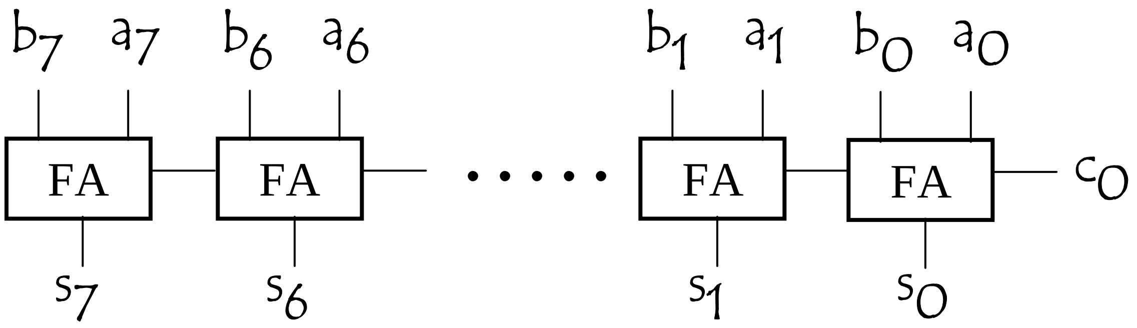
Timing should be improved

Adders

Adding two natural numbers

At each stage, I need to sum 3 single bit numbers a_i b_i c_i

The carry out of the stage i is the input carry of the next stage



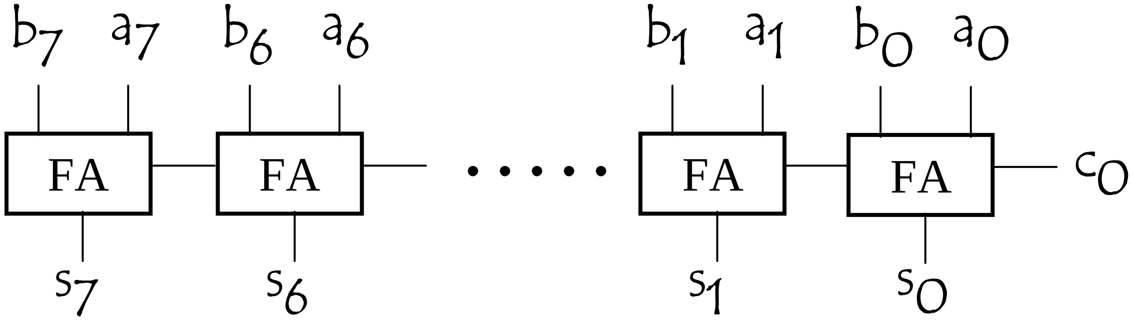
Ripple Carry Adder (RCA)

Adders

Adding two natural numbers

Ripple Carry Adder (RCA)

Area ∝ n   Delay ∝ n

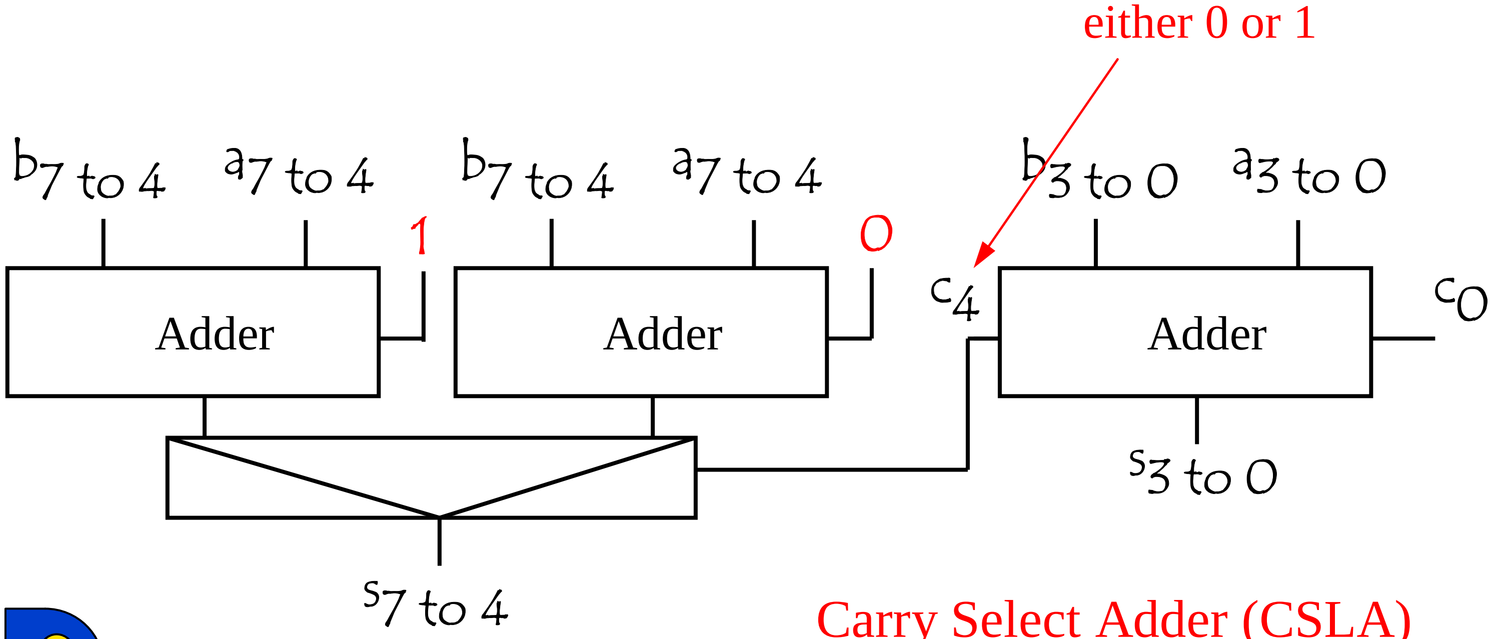


Timing should be improved

Adders

Adding two natural numbers

Acceleration technics



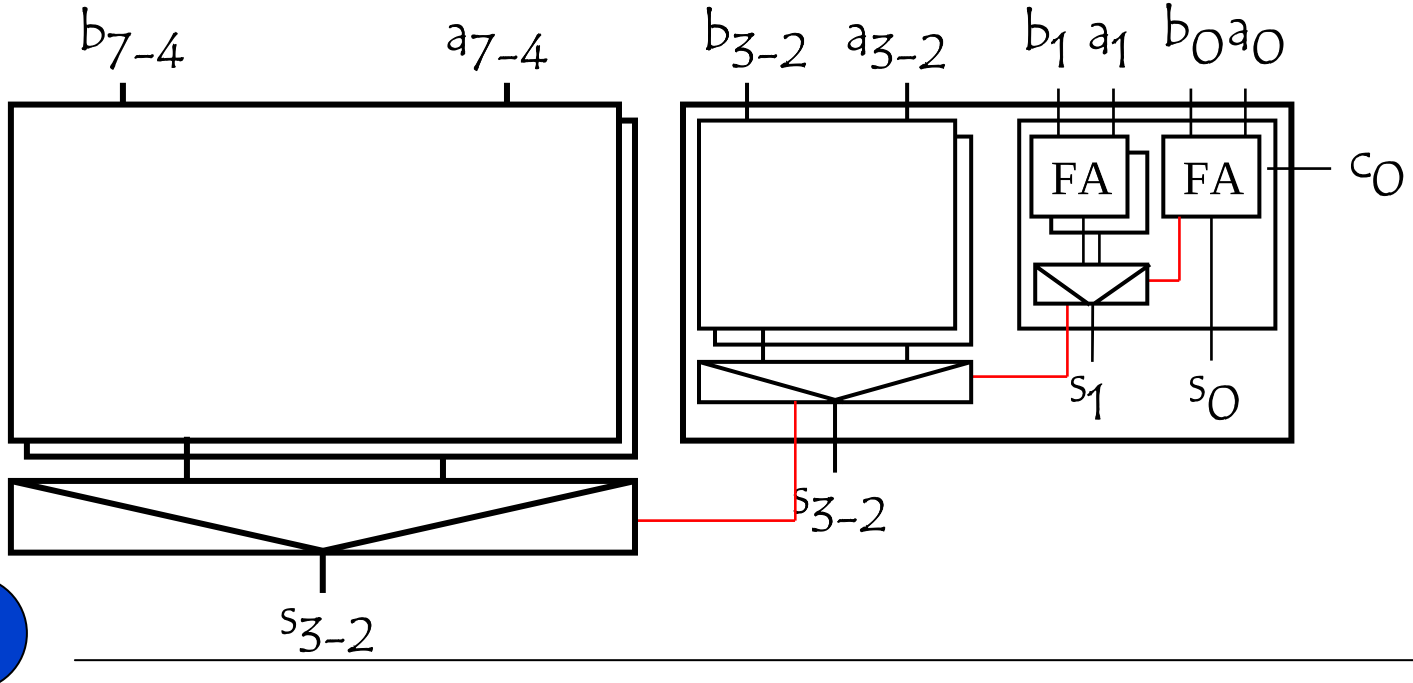
Carry Select Adder (CSLA)

Adders

Adding two natural numbers

Carry Select Adder (CSLA)

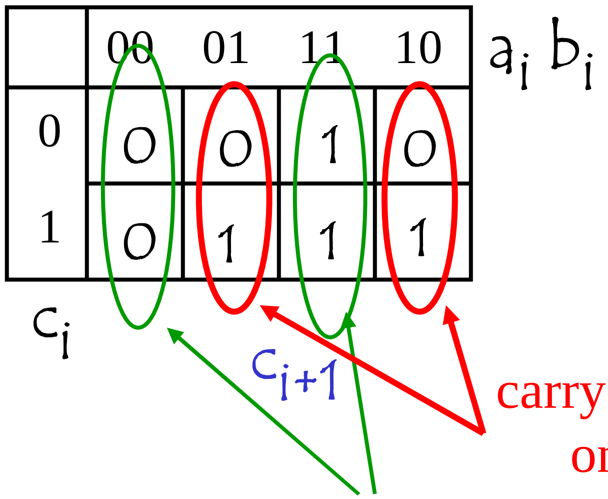
Area $\propto n^{log(3)} = n^{1.585}$ Delay $\propto$ *log* (n)



Adders

Adding two natural numbers

Acceleration technics



c_i \ $a_i\ b_i$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

c_{i+1}

carry out depends on carry in

carry out does not depend on carry in

00	01	11	10
absorption	propagation	generation	propagation

$a_i\ b_i$

Adders

Adding two natural numbers

Acceleration technics

	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$a_i\ b_i$

c_i

c_{i+1}

$$G_i = a_i\, b_i$$

$$P_i = a_i \oplus b_i$$

$$c_{i+1} = G_i + P_i\, c_i$$

$$s_i = P_i \oplus c_i$$

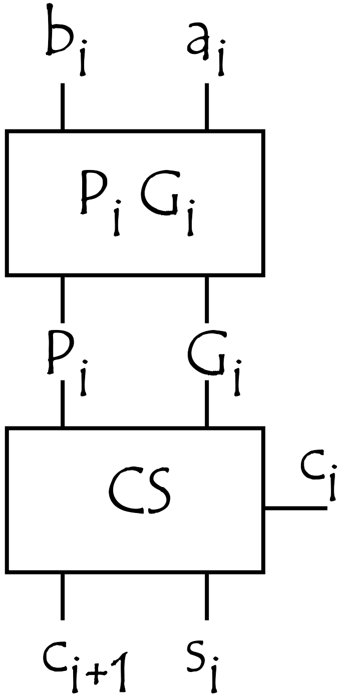
00	01	11	10
absorption	propagation	generation	propagation

$a_i\ b_i$

Adders

Adding two natural numbers

Acceleration technics



$$G_i = a_i\, b_i$$

$$P_i = a_i \oplus b_i$$

$$c_{i+1} = G_i + P_i\, c_i$$

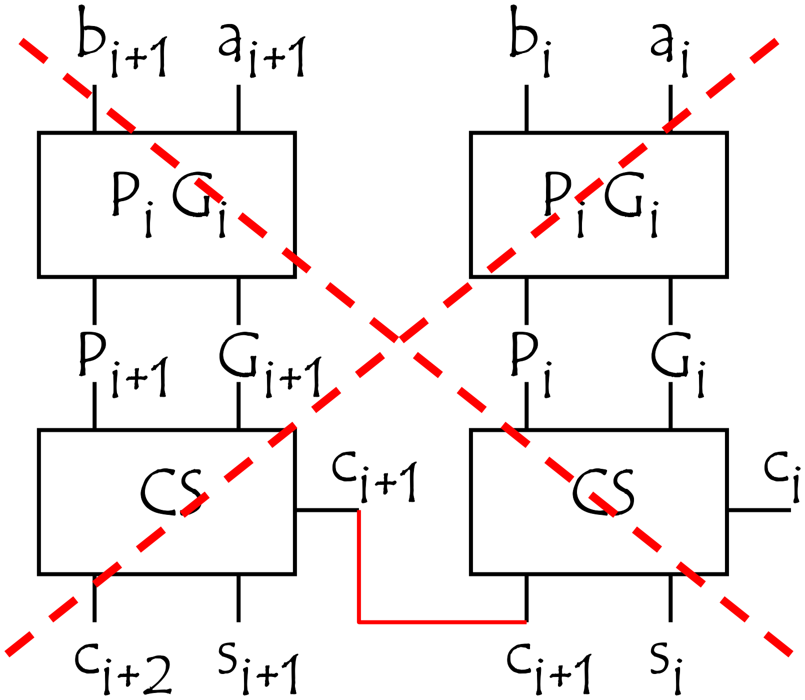
$$s_i = P_i \oplus c_i$$

00	01	11	10	$a_i\ b_i$
absorption	propagation	generation	propagation	

Adders

Adding two natural numbers

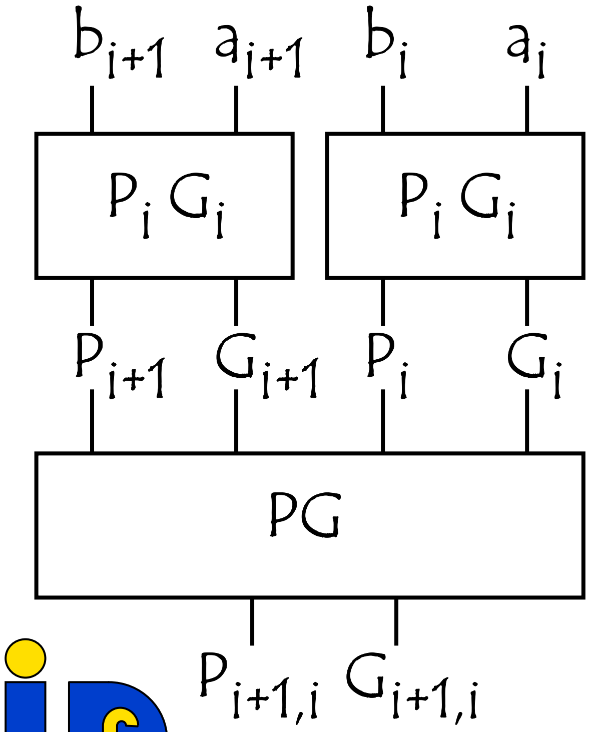
Acceleration technics



Adders

Adding two natural numbers

Acceleration technics


b_{i+1} a_{i+1} b_i a_i
$P_i\,G_i$
$P_i\,G_i$
P_{i+1} G_{i+1} P_i G_i
PG
$P_{i+1,i}$ $G_{i+1,i}$

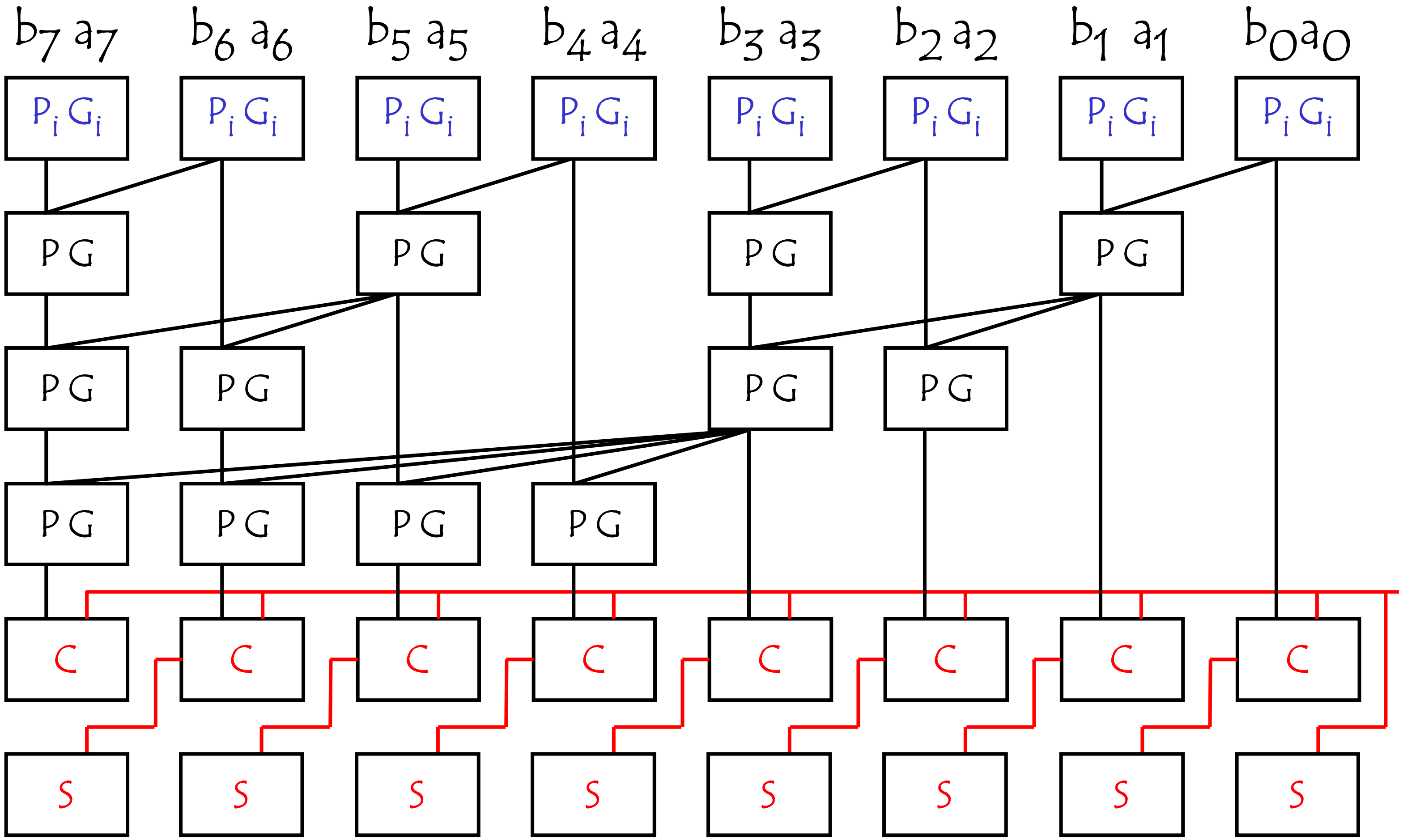
$G_{i+1} = a_{i+1}\, b_{i+1}$ $\quad G_i = a_i\, b_i$

$P_{i+1} = a_{i+1} \oplus b_{i+1}$ $\quad P_i = a_i \oplus b_i$

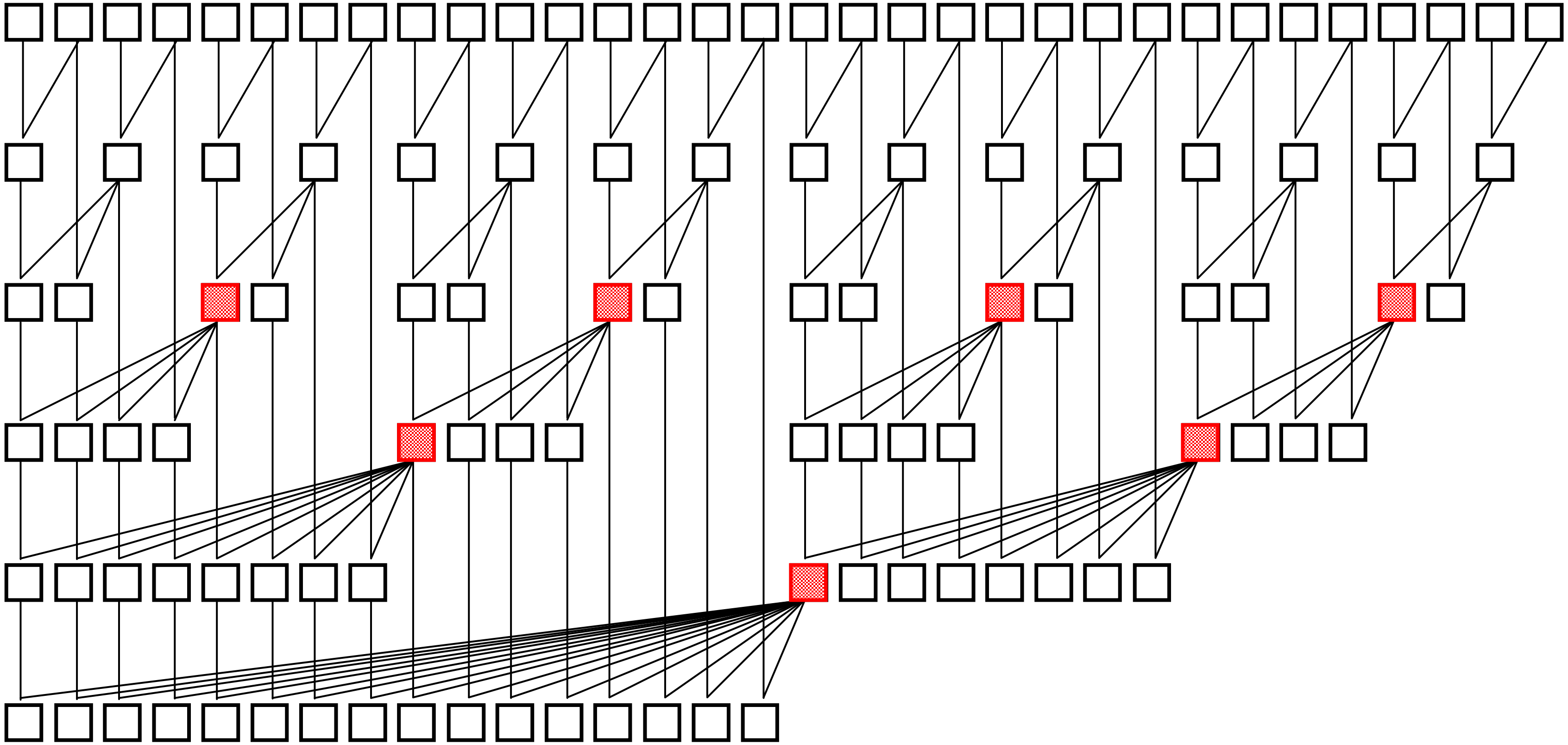
$G_{i+1,\,i} = G_{i+1} + G_i.P_{i+1}$

$P_{i+1,\,i} = P_i\,.\,P_{i+1}$

Adders

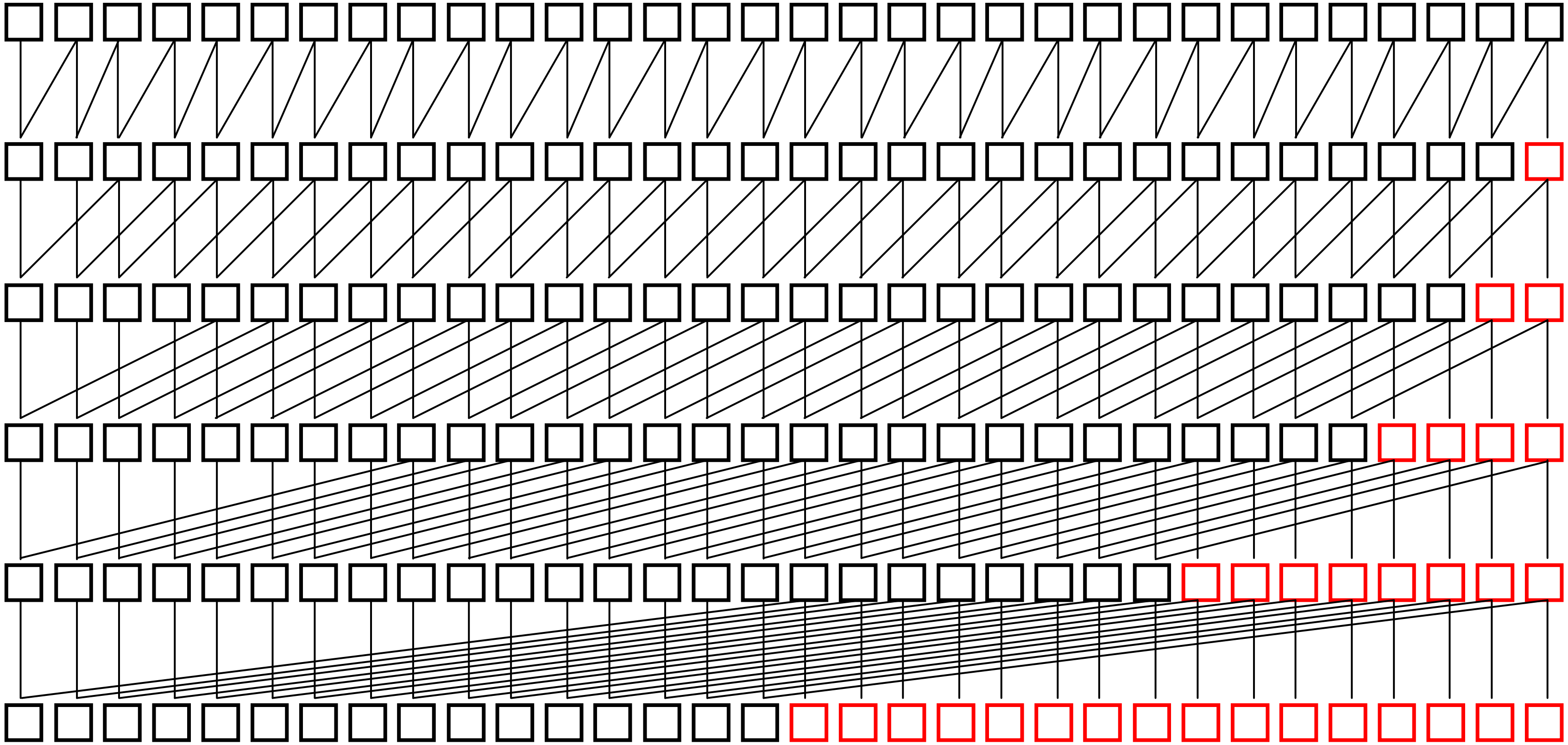


Adders



Slansky Adder

Adders



Kogge-Stone Adder

Adders

Adding two natural numbers (summary)

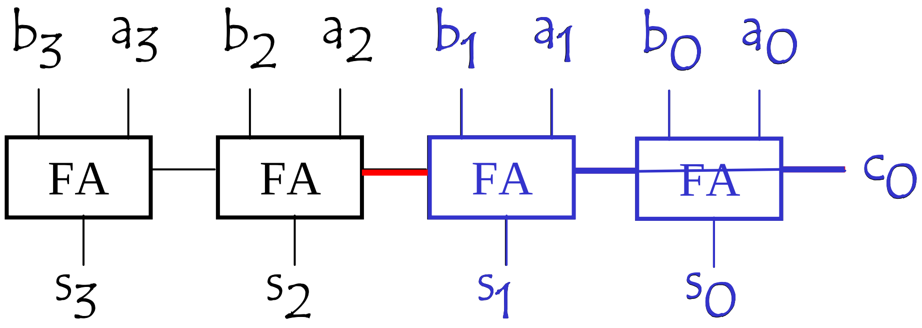
	Area	Delay
Ripple Carry (RCA)	n	n
Carry Select (CSLA)	$n^{\log(3)}$	$\log(n)$
Carry Lookahead (CLA)	$n \log(n)$	$\log(n)$
Magic Adder	n	Cste

More improvements ?

Adders

Adding two natural numbers

Ripple Carry Adder (RCA)

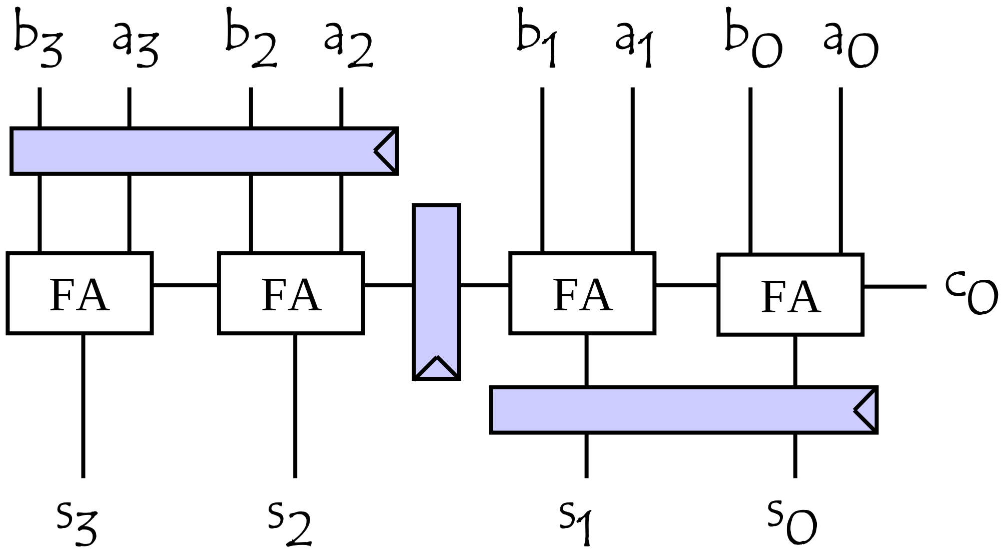


Adders

Adding two natural numbers

Ripple Carry Adder (RCA)

pipelining

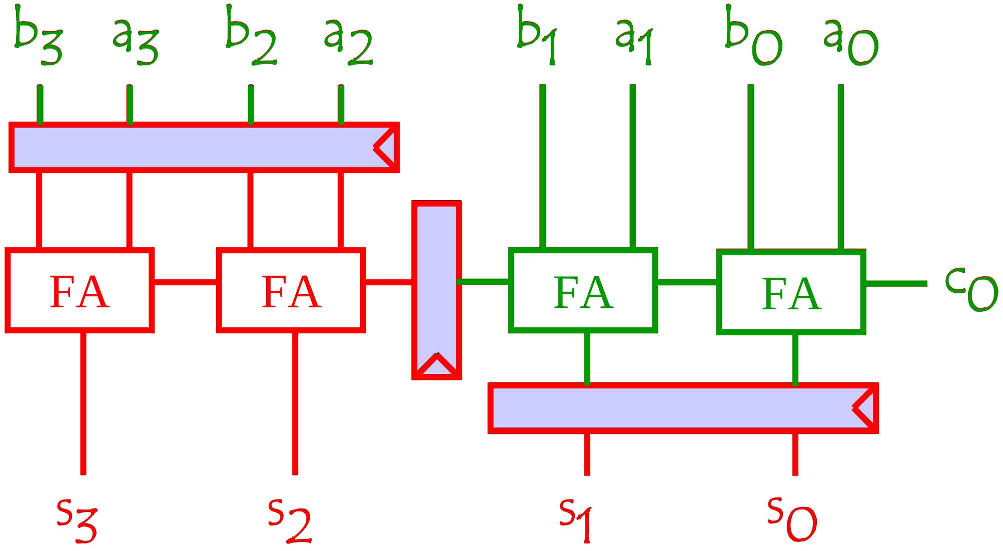


Adders

Adding two natural numbers

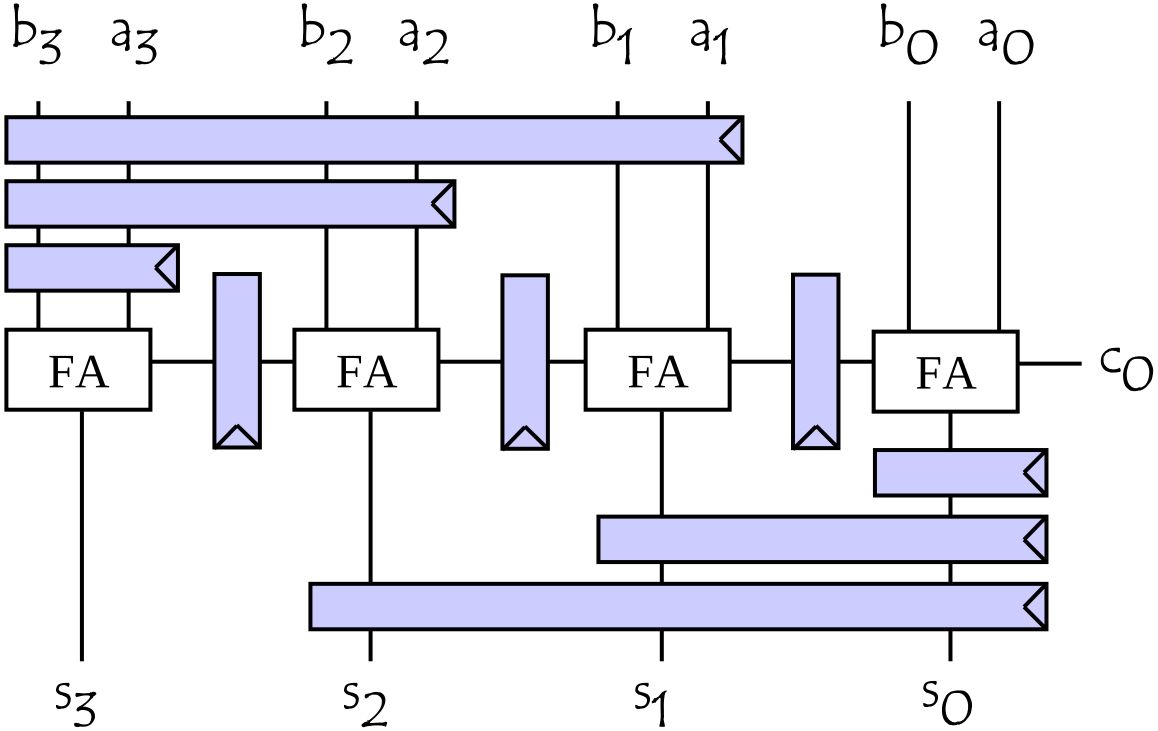
Ripple Carry Adder (RCA)

pipelining



Adders

Adding two natural numbers



Adders

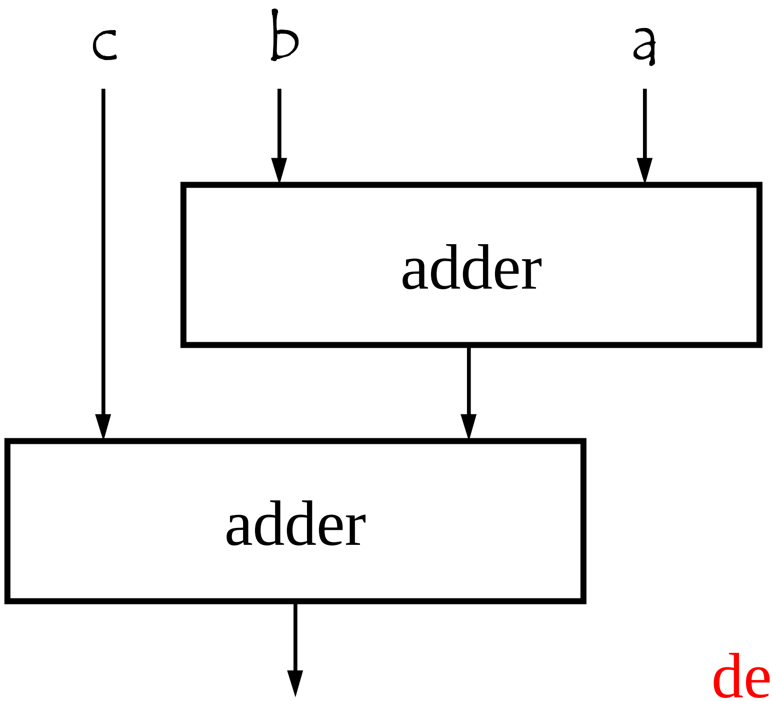
Adding two natural numbers (summary)

	Area	Delay
Ripple Carry (RCA)	n	n
Carry Select (CSLA)	$n^{\mathit{log}(3)}$	*log* (n)
Carry Lookahead (CLA)	n *log* (n)	*log* (n)
Pipeline Adder	n^{2}	Cste (1 cycle)
Magic Adder	n	Cste

When there is no door to escape break the wall

Adders

Adding three natural numbers



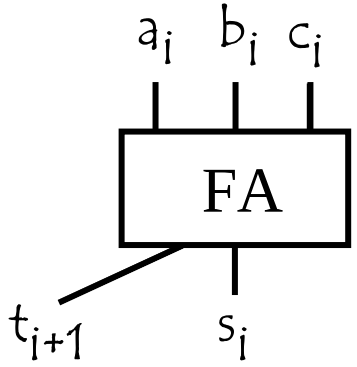
delay

Adders

Adding three natural numbers

$s_i = a_i \oplus b_i \oplus c_i$ $c_{i+1} = a_i.b_i + a_i.c_i + b_i.c_i$

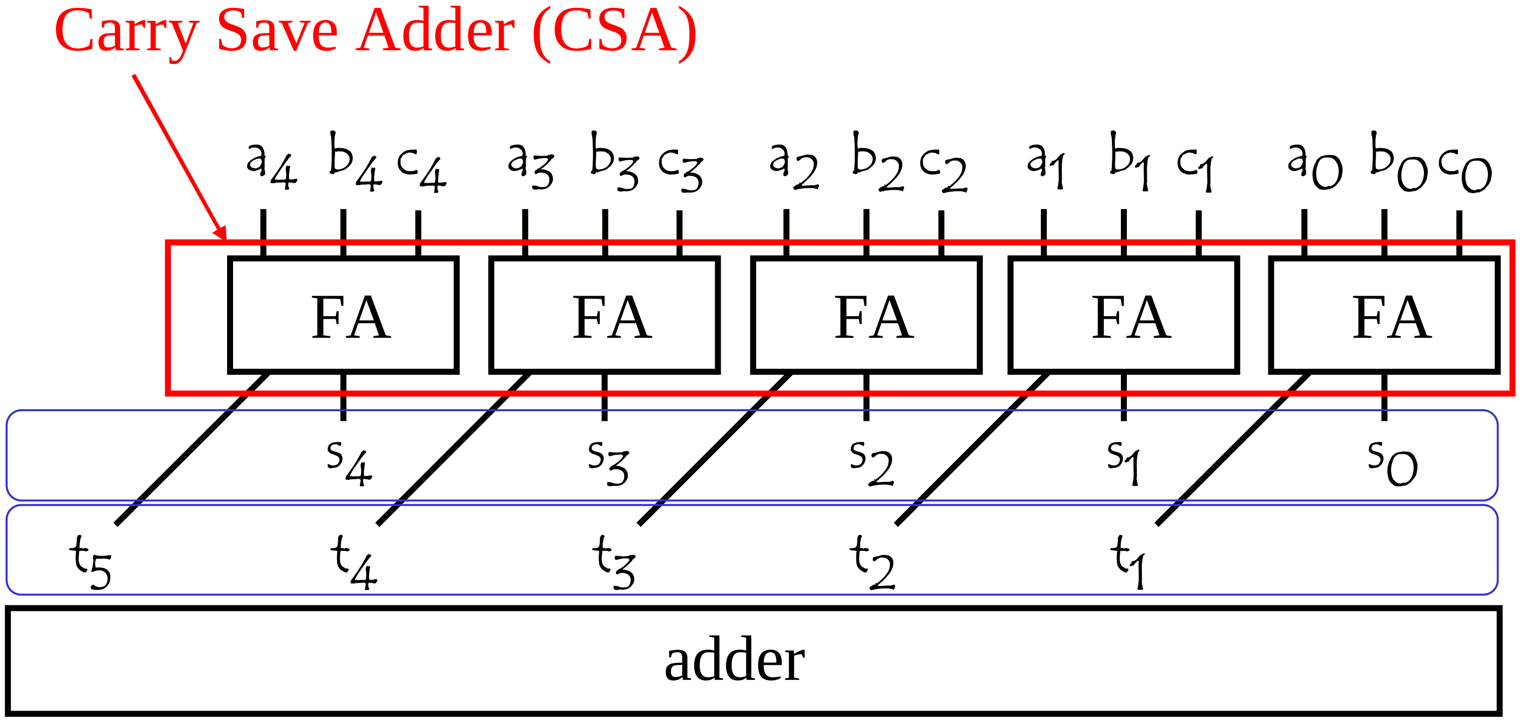
the expressions are symmetrical in regard of a, b and c



A full adder creates 2 numbers from 3

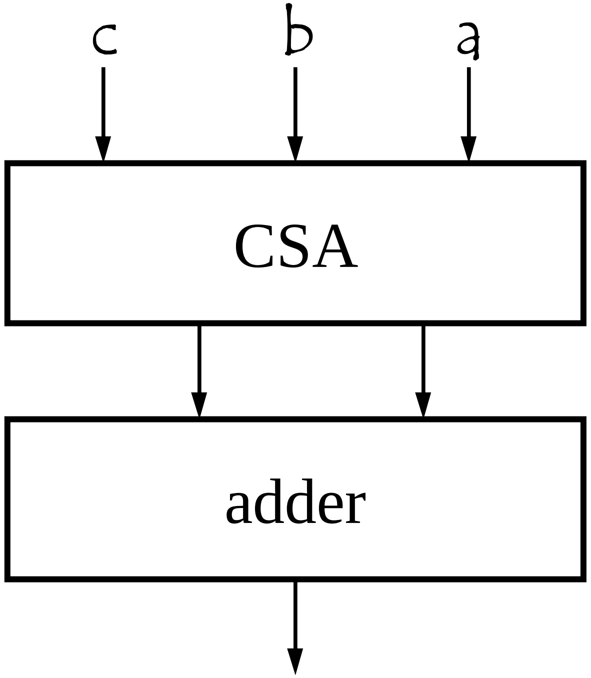
Adders

Adding three natural numbers



Adders

Adding three natural numbers



Delay = cste

Area $\propto$ n

Adders

Adding two natural numbers

Change the representation of numbers

Given a natural number a : a is coded using 2n bits

$a = a0 + a1$ Redundant Binary Code

Example : the number 5 can be coded on 4 bits as

0000 + 0101

0001 + 0100

0010 + 0011

Adders

Adding two natural numbers

Changing the representation of numbers

$$a = a0 + a1$$

$$b = b0 + b1$$

Adding a and b in Redundant Binary Code is finding c

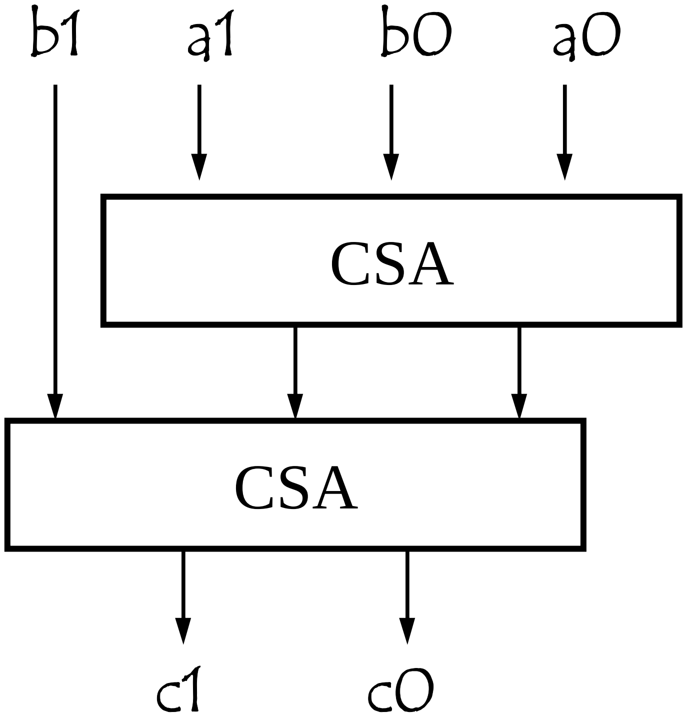
$c = c0 + c1$ such as

$$c0 + c1 = a0 + a1 + b0 + b1$$

Adding 4 numbers to generate 2

Adders

Adding two natural numbers



Delay = cste

Area $\propto$ n