

# Combinatorial Circuits

Networks and Embedded Software

Module 3.3.1

by Wolfgang Neff

# Design (1)

- Find a digital circuit for a given problem

- Please add two bits.

If you add two numbers a carry can occur. You need not only determine the result but also the carry. On the other hand there can already be a carry when you add two numbers. You have to handle this carry, too. This type of circuit is called full adder.

- Mathematical description of a full adder

- $\{0,1\} \times \{0,1\} \times \{0,1\} \mapsto \{0,1\} \times \{0,1\}$

- $(a,b,c_i) \rightarrow (c_o,r)$

- $c_i$ : carry in;  $c_o$ : carry out;  $r$ : LSB of  $a+b$  (left most bit)

Or for short:  
 $\{0,1\}^3 \mapsto \{0,1\}^2$

# Design (2)

- Example

Bit position	3	2	1	0
1st Number (7)	0	1	1	1
2nd Number (5)	0	1	0	1
Carry	1	1	1	0
Result (12)	1	1	0	0

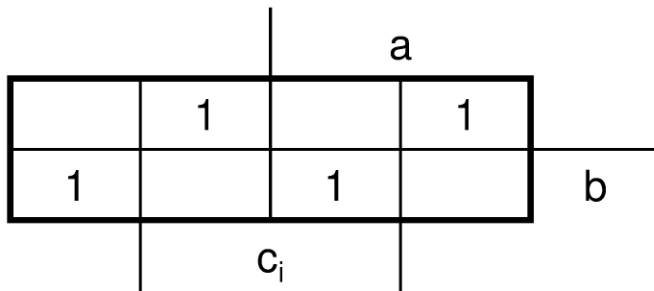
# Design (3)

- Truth table of a one bit full adder

a	b	$c_i$	$c_o$	y
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

# Design (4)

- Switching function  $y(a,b,c_i)$



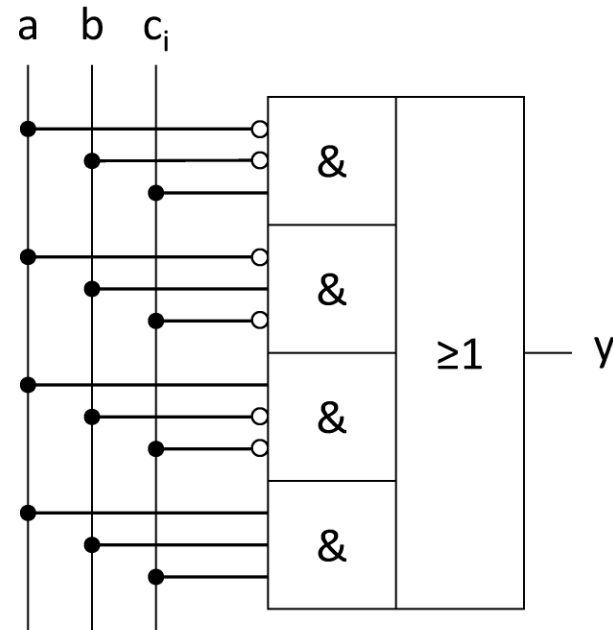
$$t_0 = \neg a \wedge \neg b \wedge c_i$$

$$t_1 = \neg a \wedge b \wedge \neg c_i$$

$$t_2 = a \wedge \neg b \wedge \neg c_i$$

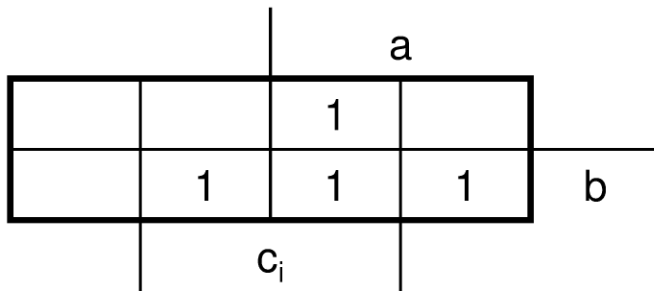
$$t_3 = a \wedge b \wedge c_i$$

$$y = t_0 \vee t_1 \vee t_2 \vee t_3$$



# Design (5)

- Switching function  $c_o(a,b,c_i)$

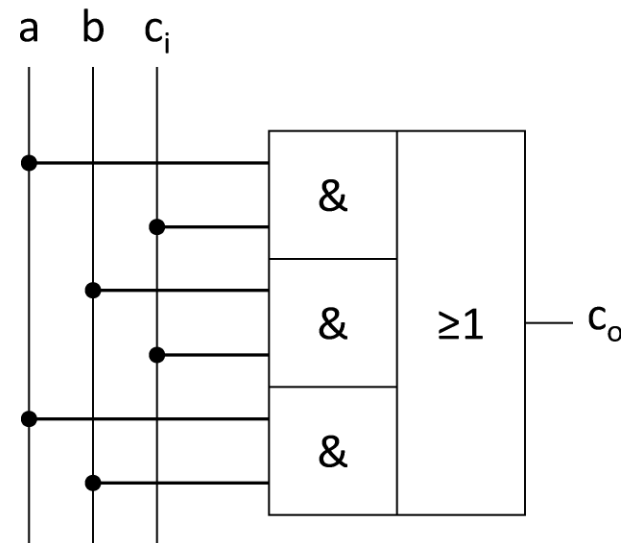


$$t_0 = a \wedge c_i$$

$$t_1 = b \wedge c_i$$

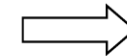
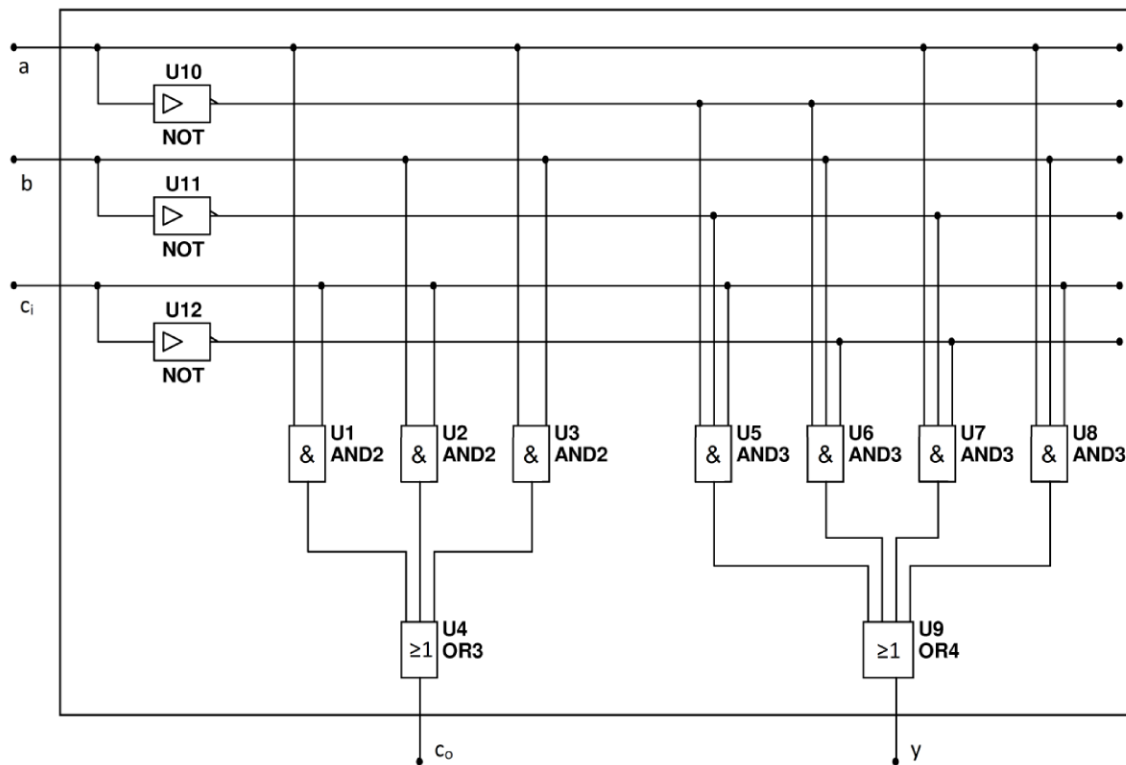
$$t_2 = a \wedge b$$

$$c_o = t_0 \vee t_1 \vee t_2$$

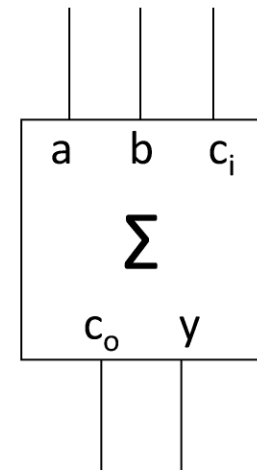


# Design (6)

- Both functions integrated in a circuit

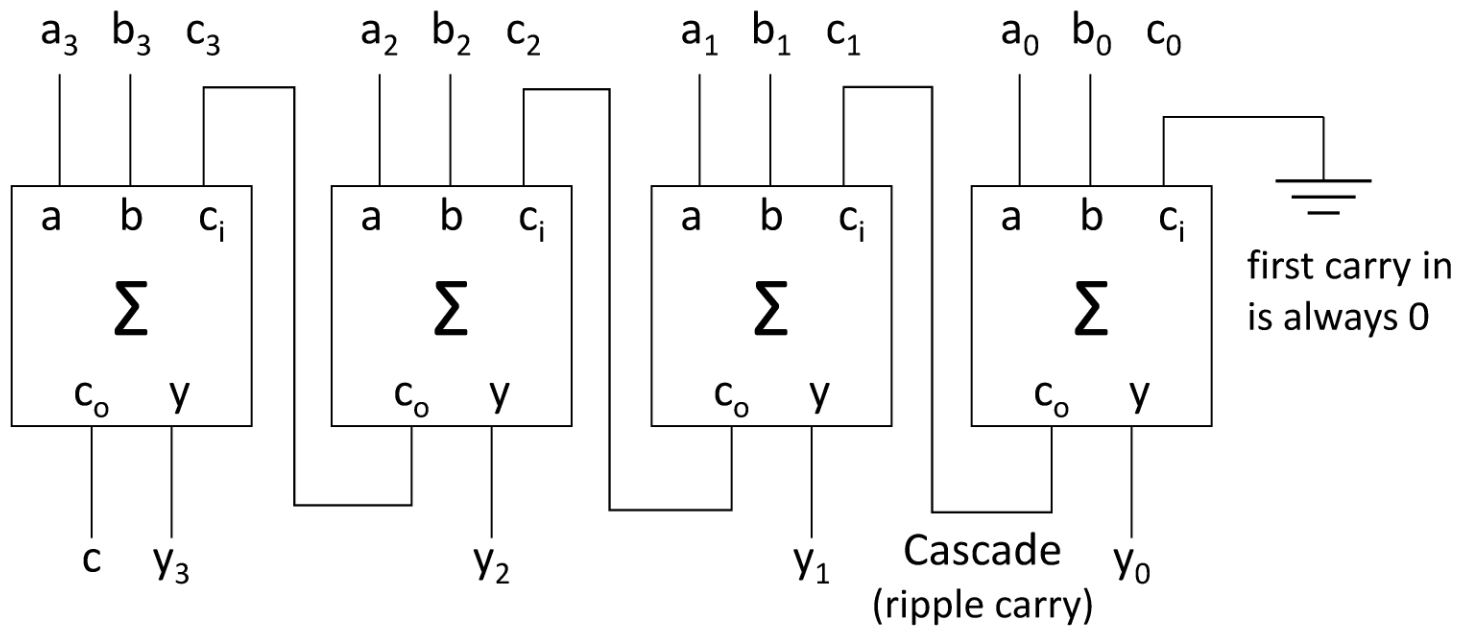


One-Bit Full Adder



# Design (7)

- Four-bit Full Adder
  - Cascade of four one-bit full adders





# Design (8)

- Integrated four bit half adder

