USP - ICMC - SSC
SSC 0113 (Lab ELD II) - 2o. Semestre 2012

Disciplina de
SSC0113 - Elementos de Lógica Digital II (Prática)

Prof. Fernando Osório
Email: fosorio [at] { icmc. usp. br , gmail. com }
Estagiário PAE: Diogo Ortiz Correa
Email: diogosoc [at] { icmc. usp. br }
Web: http://www.icmc.usp.br/~fosorio/
Wiki ICMC: http://wiki.icmc.usp.br/index.php/SSC-113-2012(fosorio)

Aula 03

Aula 02 – Circuitos Lógicos e VHDL

Agenda:

1. Somador em FPGA com VHDL

   Somador: Implementação Xor , Half-Adder , Full-Adder

2. Contador: Display 7 Segmentos

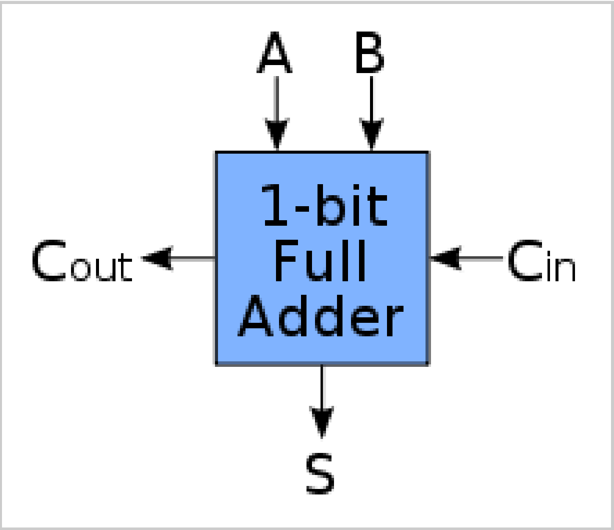
   Contador (tipo "PC")

   Exibir no Display de 7 Segmentos

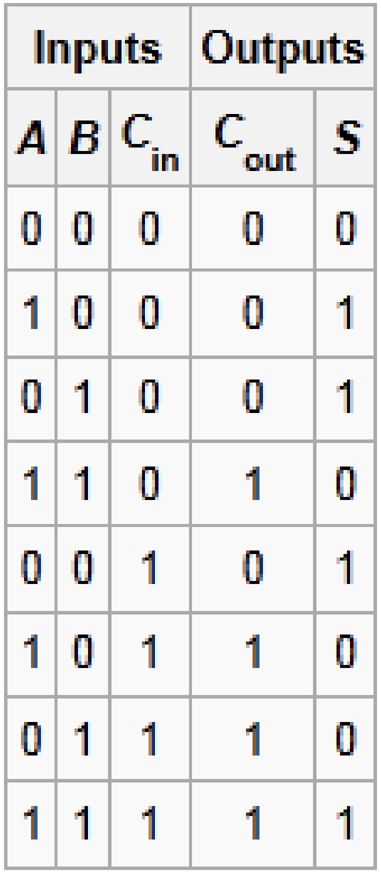
1. Projeto de Somador em FPGA

Circuitos Lógicos

- **Full-Adder**



Schematic symbol for a 1-bit full adder with C_{in} and C_{out} drawn on sides of block to emphasize their use in a multi-bit adder

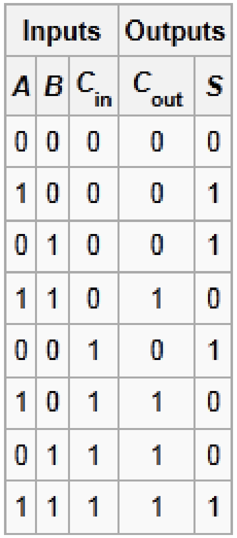


Inputs			Outputs	
A	*B*	C_{in}	C_{out}	*S*
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
1	1	0	1	0
0	0	1	0	1
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1

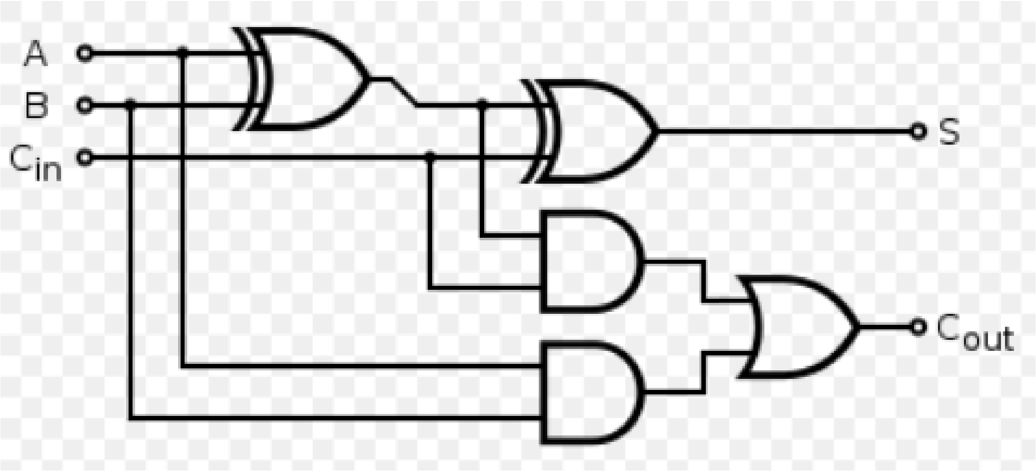
1. Projeto de Somador em FPGA

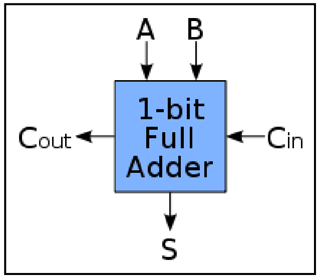
Circuitos Lógicos

- **Full-Adder**



Inputs			Outputs	
A	*B*	C_{in}	C_{out}	*S*
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
1	1	0	1	0
0	0	1	0	1
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1





1. Projeto de Somador em FPGA

VHDL

Full-Adder

```
-- Alunos: Danilo Athayde, Lucas Wiechmann,
--         Rodrigo Pereira

library IEEE;
use IEEE.Std_Logic_1164.all;

ENTITY somador IS
   PORT( A, B, Cin : IN STD_LOGIC;
         Cout, S   : OUT STD_LOGIC );
END somador;

ARCHITECTURE Structure OF somador IS
BEGIN
  S <= ( (A XOR B) XOR Cin );
  Cout <= ( (A AND B) OR (A AND Cin) OR (B AND Cin) );
END Structure;
```

1. Projeto de Somador em FPGA

VHDL

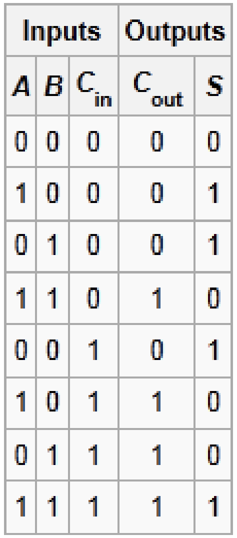
Full-Adder

```
-- Alunos: Danilo Athayde, Lucas Wiechmann,
--         Rodrigo Pereira

library IEEE;
use IEEE.Std_Logic_1164.all;

ENTITY somador IS
   PORT( A, B, Cin : IN STD_LOGIC;
         Cout, S   : OUT STD_LOGIC );
END somador;

ARCHITECTURE Structure OF somador IS
BEGIN
  S <= ( (A XOR B) XOR Cin );
  Cout <= ( (A AND B) OR (A AND Cin) OR (B AND Cin) );
END Structure;
```

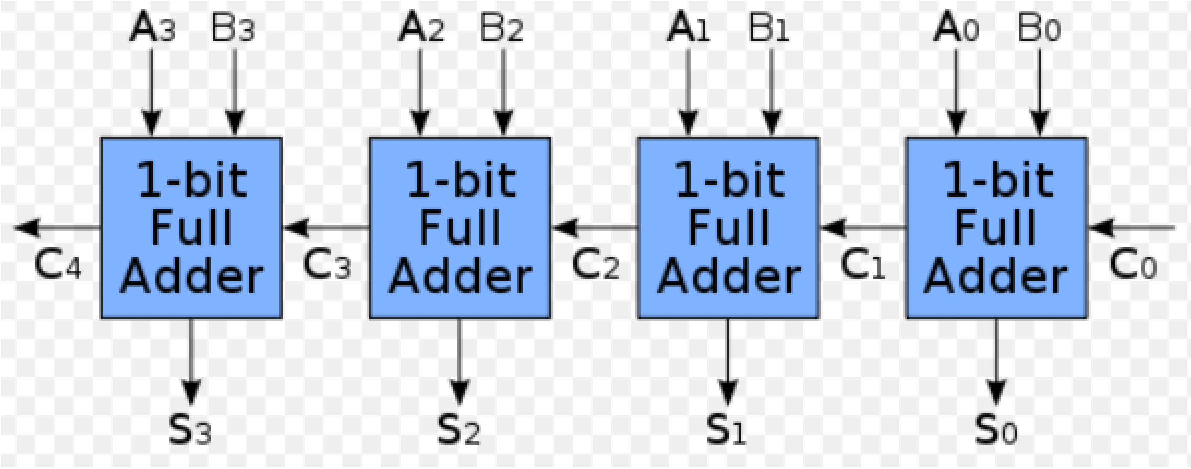
1. Projeto de Somador em FPGA

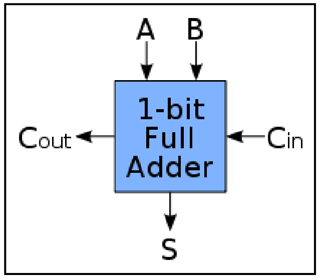
Circuitos Lógicos

- **Full-Adder em Cascata: Somador de 4 bits**



Inputs			Outputs	
A	*B*	C_{in}	C_{out}	*S*
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
1	1	0	1	0
0	0	1	0	1
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1





Referência – Figuras:
http://en.wikipedia.org/wiki/Adder_(electronics)

1. Projeto de Somador em FPGA

VHDL

Cascade Adder

```
ENTITY somador_4bits IS
    PORT( X, Y  : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
          SCin  : IN STD_LOGIC;
          SCout : OUT STD_LOGIC;
          Z     : OUT STD_LOGIC_VECTOR(3 DOWNTO 0) );
END somador_4bits;

ARCHITECTURE Structure OF somador_4bits IS
    COMPONENT somador
        PORT( A, B, Cin       : IN STD_LOGIC;
              Cout, S         : OUT STD_LOGIC );
    END COMPONENT;
    SIGNAL carry : STD_LOGIC_VECTOR(3 DOWNTO 0);
BEGIN
 X0: somador port map( X(0), Y(0), SCin, carry(0), Z(0) );
 X1: somador port map( X(1), Y(1), carry(0), carry(1), Z(1) );
 X2: somador port map( X(2), Y(2), carry(1), carry(2), Z(2) );
 X3: somador port map( X(3), Y(3), carry(2), SCout, Z(3) );
END Structure;
```

1. Projeto de Somador em FPGA

```
ENTITY somador_4bits IS
  PORT( X, Y: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        SCin: IN STD_LOGIC;
        SCout: OUT STD_LOGIC;
        Z : OUT STD_LOGIC_VECTOR(3 DOWNTO 0)
END somador_4bits;

ARCHITECTURE Structure OF somador_4bits IS
     COMPONENT somador
          PORT( A, B, Cin          : IN STD_LOGIC;
                Cout, S            : OUT STD_LOGIC );
     END COMPONENT;
     SIGNAL carry : STD_LOGIC_VECTOR(3 DOWNTO 0);
BEGIN
  X0: somador PORT MAP( X(0), Y(0), SCin, carry(0), Z(0) );
  X1: somador PORT MAP( X(1), Y(1), carry(0), carry(1), Z(1) );
  X2: somador PORT MAP( X(2), Y(2), carry(1), carry(2), Z(2) );
  X3: somador PORT MAP( X(3), Y(3), carry(2), SCout, Z(3) );
END Structure;
```

```
library IEEE;
use IEEE.Std_Logic_1164.all;

ENTITY somador IS
    PORT( A, B, Cin : IN STD_LOGIC;
          Cout, S   : OUT STD_LOGIC );
END somador;

ARCHITECTURE Structure OF somador IS
BEGIN
  S <= ( (A XOR B) XOR Cin );
  Cout <= ( (A AND B) OR (A AND Cin) OR (B AND Cin) );
END Structure;
```

1. Projeto de Somador em FPGA

VHDL

Adder

Behavior

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
ENTITY addsub IS
    PORT ( a:   IN INTEGER;
           b:   IN INTEGER;
           addnsub: IN BOOLEAN;
           result:  OUT INTEGER   );
END addsub;
ARCHITECTURE Behavior OF addsub IS
BEGIN
    PROCESS (a, b, addnsub)
    BEGIN
        IF (addnsub) THEN
           result <= a + b;
        ELSE
           result <= a - b;
        END IF;
    END PROCESS;
END Behavior;
```

1. Projeto de Somador em FPGA

VHDL

Adder

Behavior

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
ENTITY addsub IS
    PORT ( a:   IN INTEGER;
           b:   IN INTEGER;
           addnsub: IN BOOLEAN;
           result:  OUT INTEGER   );
END addsub;
ARCHITECTURE Behavior OF addsub IS
BEGIN
    PROCESS (a, b, addnsub)
    BEGIN
        IF (addnsub) THEN
           result <= a + b;
        ELSE
           result <= a - b;
        END IF;
    END PROCESS;
END Behavior;
```

VHDL

Adder Generic

```
-- From: http://www.csee.umbc.edu/portal/help/VHDL/samples/add_g.vhdl
-- Adapted for use with Quartus from “add_g.vhdl” by FSO

library IEEE;
use IEEE.std_logic_1164.all;

entity add_g is

  generic(left : natural := 31 );        -- top bit

    port (a    : in  std_logic_vector (left downto 0);
          b    : in  std_logic_vector (left downto 0);
          cin  : in  std_logic;
          sum  : out std_logic_vector (left downto 0);
          cout : out std_logic);
end entity add_g;

architecture behavior of add_g is

begin  -- behavior
adder: process (a, b)
         variable carry : std_logic; -- internal
         variable isum : std_logic_vector(left downto 0);   -- internal
begin
    carry := cin;
    for i in 0 to left loop
        isum(i)  := a(i) xor b(i) xor carry;
        carry  := (a(i) and b(i)) or (a(i) and carry) or (b(i) and carry);
     end loop;
     sum  <= isum;
     cout <= carry;
end process adder;

end architecture behavior;  -- of add_g
```

VHDL

Adder

Generic

```
-- From: http://www.csee.umbc.edu/portal/help/VHDL/samples/add_g.vhdl
-- Adapted for use with Quartus from “add_g.vhdl” by FSO

library IEEE;
use IEEE.std_logic_1164.all;

entity add_g is

  generic(left : natural := 31 );          -- top bit

    port (a     : in  std_logic_vector (left downto 0);
          b     : in  std_logic_vector (left downto 0);
          cin   : in  std_logic;
          sum   : out std_logic_vector (left downto 0);
          cout  : out std_logic);
end entity add_g;

architecture behavior of add_g is

begin  -- behavior
adder: process (a, b)
         variable carry : std_logic; -- internal
         variable isum : std_logic_vector(left downto 0);  -- internal
begin
    carry := cin;
    for i in 0 to left loop
        isum(i) := a(i) xor b(i) xor carry;
        carry  := (a(i) and b(i)) or (a(i) and carry) or (b(i) and carry);
     end loop;
     sum  <= isum;
     cout <= carry;
end process adder;

end architecture behavior;  -- of add_g
```

2. Projeto de Somador em FPGA

VHDL: Adder - REFERENCIAS

SITES:

AULA 02 => http://osorio.wait4.org/SSC0113/AULA02/

VHDL Reference => http://osorio.wait4.org/SSC0113/VHDL/ **(Livro B&V)**

EECS 316 – Lecture 4: The VHDL N-bit adder:
http://bear.cwru.edu/eecs_316/eecs_317_20010209.ppt
http://osorio.wait4.org/SSC0113/AULA02/eecs_317_adder.pdf

ADDER Generic:
http://www.csee.umbc.edu/portal/help/VHDL/samples/samples.shtml
http://www.csee.umbc.edu/portal/help/VHDL/samples/add_g.vhdl

ADDER +1 Exemplo:
http://esd.cs.ucr.edu/labs/tutorial/adder.vhd **ou** http://esd.cs.ucr.edu/labs/tutorial/

ALTERA Examples:
http://www.altera.com/support/examples/vhdl/vhdl.html **(Adder)**

2. Projeto de Contador em FPGA

VHDL: Contador - DICAS & REFERENCIAS

>> Implementar um CONTADOR usando o CLOCK como referência e mostrando a saída no DISPLAY de 7 segmentos (2 casas em modo decimal – 0 a 9)

SITES:

AULA 03:
http://osorio.wait4.org/SSC0113/AULA03/

Livro VHDL Roberto D'Amore:
http://www.ele.ita.br/~damore/vhdl/
(exemplos – ver Cap.1)

Atribuição de Pinos FPGA – DE2-70:
http://wiki.icmc.usp.br/index.php/Arquivo:De2-70-Pins.txt
(Clk, Display, Switch)

INFORMAÇÕES SOBRE A DISCIPLINA

USP - Universidade de São Paulo - São Carlos, SP
ICMC - Instituto de Ciências Matemáticas e de Computação
SSC - Departamento de Sistemas de Computação

LRM – Laboratório de Robótica Móvel
Web LRM: Http://lrm.icmc.usp.br/
Página pessoal: Http://www.icmc.usp.br/ ~fosorio/
E-mail: fosorio [at] { icmc. usp. br , gmail. com } – F.Osório
E-mail: diogosoc [at] { icmc. usp. br } - Diogo Correa (PAE)

Disciplina de Laboratório de Elementos de Lógica Digital II [LELD2]
Web Disciplinas: Http://www.icmc.usp.br/~fosorio/
Web Wiki: http://wiki.icmc.usp.br/index.php/SSC-113-2012(fosorio)
> Programa, Material de Aulas, Critérios de Avaliação,
> Material de Apoio, Trabalhos Práticos