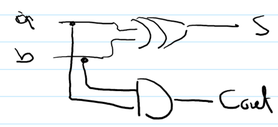
Structural vhdl

12 October 2021 18:31

VHDL

behavioural — XOR, AND, OR — "simple"

↳ Half adder



S <= a xor b ;

Cout <= a and b ;

entity port.

inputs & output

architecture

behave

structural

HA → FA → 4 bit Add → BCD Adder

entity

input

output

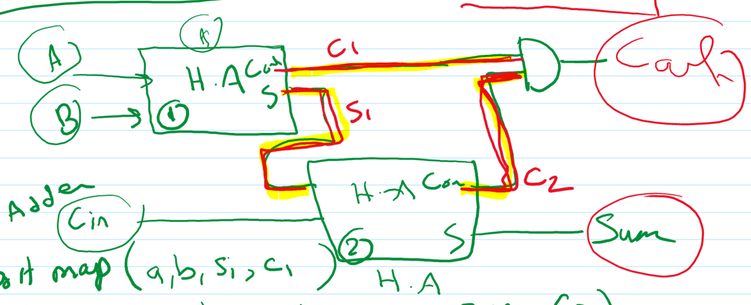
architecture

Component HALF_Adder

entity a, b → input

signals s_1, s_2, s_3, c_1, c_2

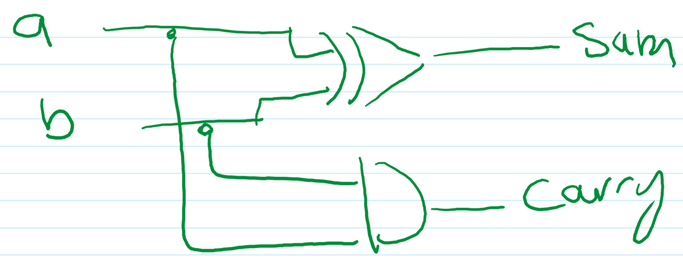
port map



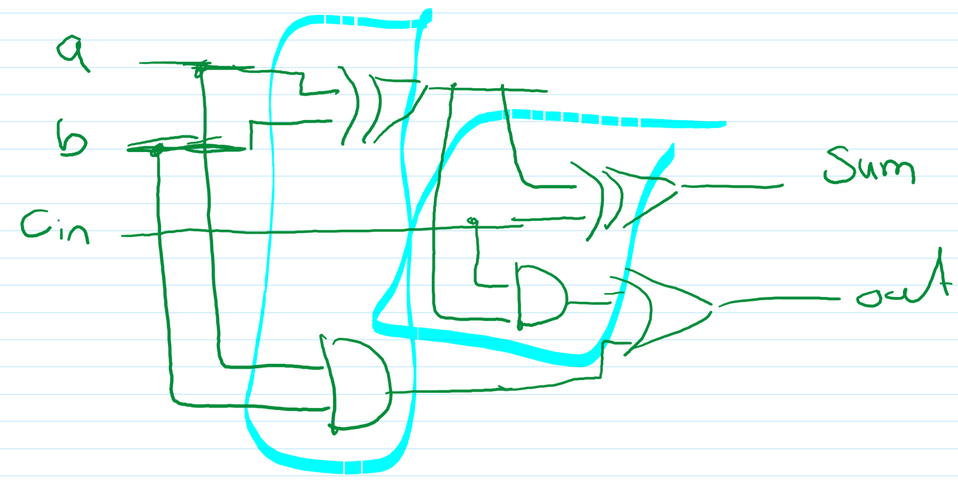
H.A1 : HALF_Adder port map (a, b, s_1, c_1)

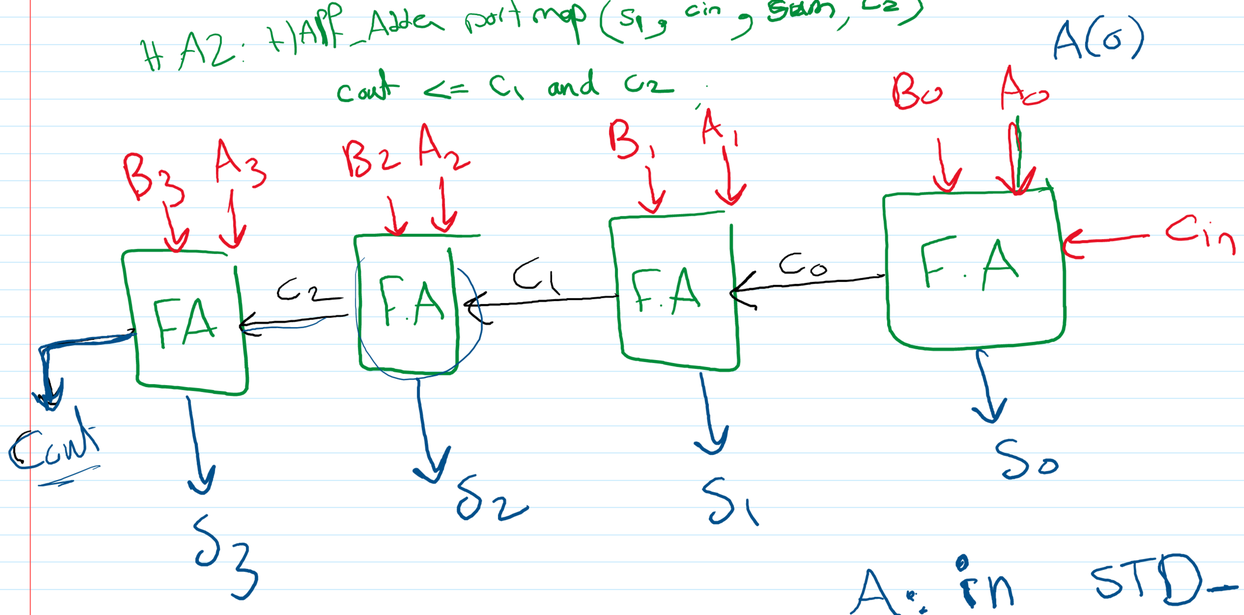
H.A2 : HALF_Adder port map (s_1, cin, Sum, c_2)

Cout <= c_1 and c_2



Half_Adder_beh.





- Signals
- inputs
- outputs

A: in STD_logic_vector (3 downto 0)

```
entity Four_bit_Adder Port(
    A: in STD_logic_vector (3 downto 0);
    B: in STD_logic_vector (3 downto 0);
    Cin: in STD_logic
    S: out STD_logic_vector (3 downto 0)
    Cout: out STD_logic;
);

architecture
    Component Full_Adder
        a, b, cin: in std_logic
        Sum, Cout: out std_logic
    signals C0, C1, C2
(0) F.A0: Full_Adder port map (A(0), B(0), Cin, S(0), C0);
(1)                           (A(1), B(1), C0, S(1), C1);
(2)                           (A(2), B(2), C1, S(2), C2);
(3)                           (A(3), B(3), C2, S(3), Cout);
end architecture
```