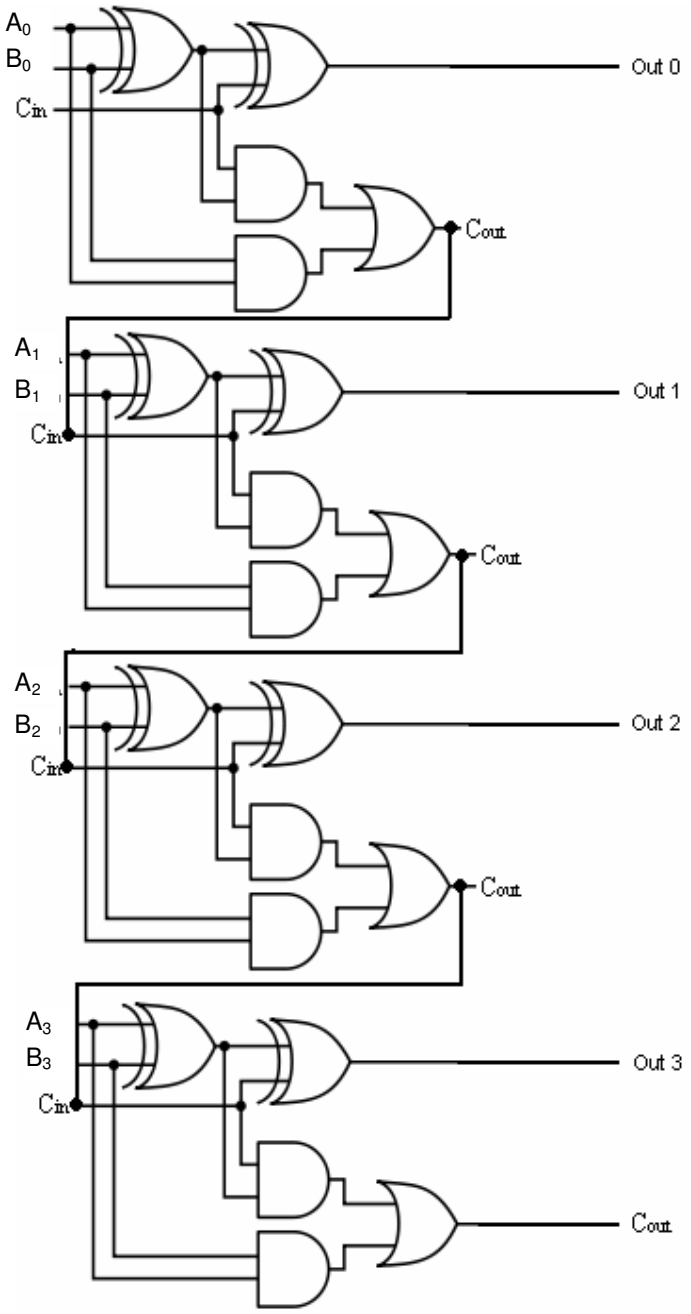
Пример 5. 4-битов суматор описан на VHDL със структурен модел

1. Логическа схема на 4-битов суматор:



Фиг.1. 4-битов суматор - A_0 е младшия бит, A_3 е старшия бит

2. Принцип на действие:

4-битовият суматор е логическа схема, която изчислява сумата на две 4-битови двоични числа.

Проектираният суматор е с последователен пренос. Той се състои от 4 последователно свързани еднобитови суматори. Първият пълен суматор е отговорен за събирането на първите две цифри на 4-битовото число, вторият за събирането на вторите две цифри, като към тях прибавя и остатъка от предния суматор и така, докато всичките битове на числото бъдат събрани.

3. Логически уравнения:

Входове: A=A3A2A1A0 Изходи: S=S3S2S1S0
B=B3B2B1B0 C_{OUT}=C4
Cin=C0

S0 = (A0 XOR B0) XOR C0;
C1 = (A0 AND B0) OR (C0 AND B0) OR (C0 AND A0)
S1 = (A1 XOR B1) XOR C1;
C2 = (A1 AND B1) OR (C1 AND B1) OR (C1 AND A1)
S2 = (A2 XOR B2) XOR C2;
C3 = (A2 AND B2) OR (C2 AND B2) OR (C2 AND A2)
S3 = (A3 XOR B3) XOR C3;
C4 = (A3 AND B3) OR (C3 AND B3) OR (C3 AND A3)

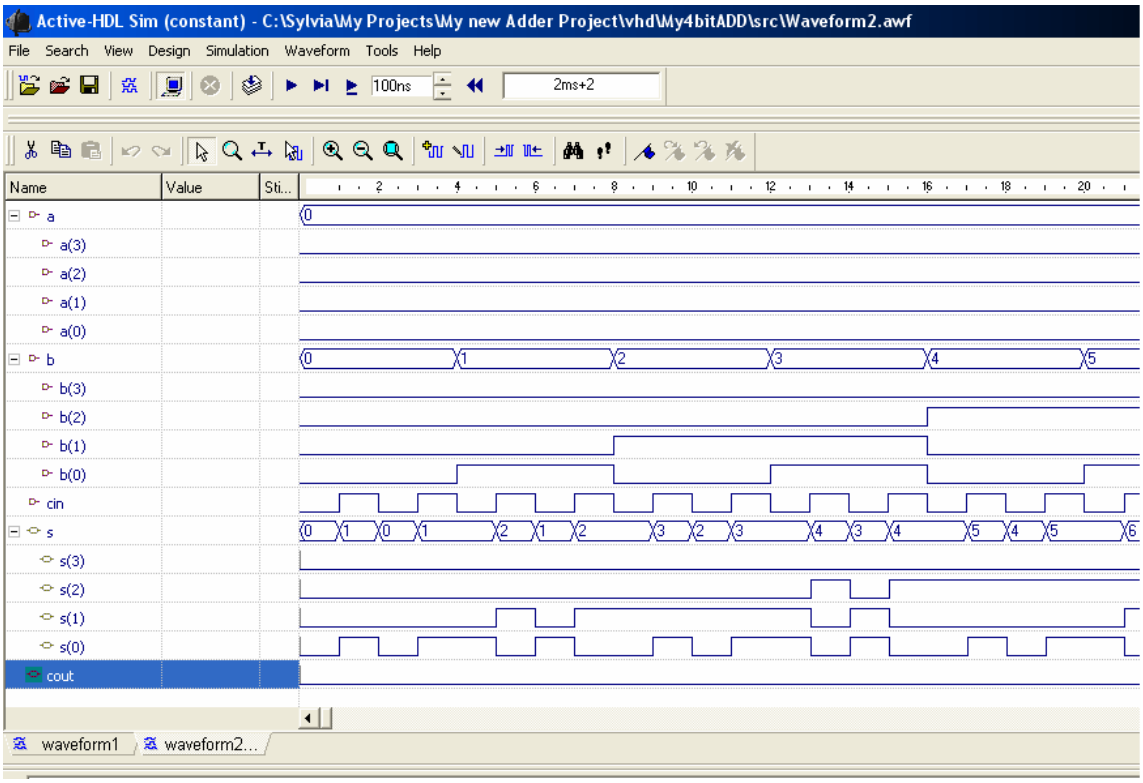
4. Описание на 4-битов суматор на VHDL със структорен модел, който използва като компонент еднобитов суматор:

```
library ieee;
use ieee.std_logic_1164.all;
entity adder_1_bit is port (a,b,Cin : in std_logic; S, Cout : out std_logic);
end adder_1_bit;
architecture structure of adder_1_bit is
begin
    process (a,b,Cin)
          begin
             S<= a xor b xor Cin;
                   Cout<= (a and b) or (b and Cin) or (a and Cin);
          end process;
end structure;
library ieee;
use ieee.std_logic_1164.all;
entity adder_4_bit is
  port ( Cin:  in std_logic;
         a,b:  in std_logic_vector(3 downto 0);
         S:    out std_logic_vector(3 downto 0);
         Cout:   out std_logic );
end adder_4_bit;
architecture structure of adder_4_bit is
  signal t: std_logic_vector (4 downto 0);
  constant n: integer:=4;
begin
 t(0) <=Cin; Cout <=t(n);
 FA_f: for i in 0 to n-1 generate
    FA_i: adder_1_bit port map (t(i), a(i), b(i), S(i), t(i+1));
 end generate;
end structure;
```

5. Входни сигнали:

Signal	Type	Frequency
A	Constant	0
b	Formulae	0.4 us 0 0.8 us 1 1.2 us 2 1.6 us 3 2 us 4 2.4 us 5
Cin	Clock	10Mhz

6. Резултати от симулацията в ACTIVE-HDL SIM:



Входните и изходните сигнали са представени като 4-битови магистрали и като десетични числа.

7. Използвани ресурси върху програмируемата схема CY37256P160-83AC и закъснения от REPORT файла adder_4_bit.rpt:

RESOURCE UTILIZATION (15:08:01)
Information: Macrocell Utilization.

Description	Used	Max
Dedicated Inputs	1	1
Clock/Inputs	4	4
I/O Macrocells	13	128
Buried Macrocells	3	128

| PIM Input Connects | 23 | 624 |

44 / 885 = 4 %

	Required	Max (Available)
CLOCK/LATCH ENABLE signals	0	20
Input REG/LATCH signals	0	133
Input PIN signals	5	5
Input PINs using I/O cells	4	4
Output PIN signals	5	124

	Required	Max (Available)
Total PIN signals	14	133
Macrocells Used	12	256
Unique Product Terms	107	1280

TIMING PATH ANALYSIS (15:08:02) using Package: cy37256p160-83ac

Messages:

```
--------------------------------------------------------------------------------
Signal Name | Delay Type |   tmax   | Path Description
--------------------------------------------------------------------------------
cmb::s(0)[42]
inp::b(0)
                    tPD          15.0 ns    1 pass
--------------------------------------------------------------------------------
cmb::s(1)[52]
inp::b(0)
                    tPD          15.0 ns    1 pass
--------------------------------------------------------------------------------
cmb::s(3)[63]
inp::b(0)
---->S_1
                    tPD          26.0 ns    2 passes
--------------------------------------------------------------------------------
cmb::s(2)[82]
inp::b(0)
---->S_4
                    tPD          26.0 ns    2 passes
--------------------------------------------------------------------------------
cmb::cout[91]
inp::b(0)
---->S_6
                     tPD          26.0 ns    2 passes
--------------------------------------------------------------------------------
Worst Case Path Summary
-----------------------
              tPD = 26.0 ns for s(3)
```