Código VHDL del sumador:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity Sumador4bits is
    Port ( a,b : in STD_LOGIC_VECTOR (3 downto 0);
           cin : in STD_LOGIC;
           S : out STD_LOGIC_VECTOR (3 downto 0);
           cout : out  STD_LOGIC);
end Sumador4bits;
architecture Behavioral of Sumador4bits is
component Sumador1Bit is
    Port ( a,b,cin : in STD_LOGIC;
           S, cout : out STD_LOGIC);
end component;
signal inversor : STD_LOGIC_VECTOR(3 DOWNTO 0);
signal cc : STD_LOGIC_VECTOR(4 downto 0);
begin
cc(0) <= cin;
ciclo : for i in 0 to 3 generate
    inversor(i) <= b(i) xor cin;
    c : Sumador1Bit port map(
        a => a(i),
        b => inversor(i),
        cin => cc(i),
        S => S(i),
        cout => cc(i+1)
    );
end generate;
cout <= cc(4);
end Behavioral;
```

Código VHDL del test - bench:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity Sumador_TB is

end Sumador_TB;
architecture Behavioral of Sumador_TB is
component Sumador4bits is
    Port ( a,b : in STD_LOGIC_VECTOR (3 downto 0);
           cin : in STD_LOGIC;
           S : out STD_LOGIC_VECTOR (3 downto 0);
           cout : out  STD_LOGIC);
end component;
signal a,b: STD_LOGIC_VECTOR(3 DOWNTO 0);
signal cin,cout : STD_LOGIC;
signal S: STD_LOGIC_VECTOR(3 DOWNTO 0);
begin
u1 : Sumador4bits
    Port Map(
    a => a,
    b => b,
    cin => cin,
    S => S,
    cout => cout
    );
    process
        begin
        a <= "0001";
        b <= "0101";
        cin <= '1';
        wait;
    end process;
end Behavioral;
```

Diagrama RTL:

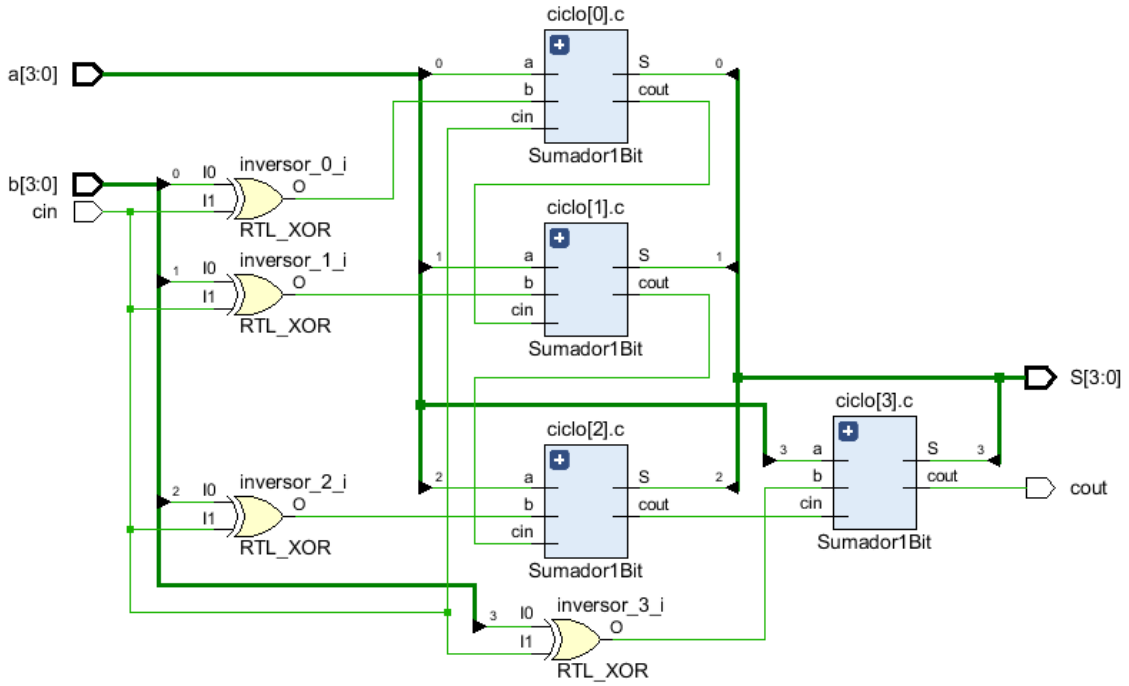

Tabla de resultados:

Operación	A	B	S	Cout
Suma	6	7	1011	0
Suma	6	9	1111	0
Suma	4	9	1101	0
Resta	15	1	1110	1
Suma	3	10	1101	0
Resta	12	5	0111	1
Resta	14	8	0110	1
Resta	10	6	0100	1
Resta	9	4	0101	1