```vhdl
-- Sumador completo BCD de 1 dígito


library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity sum_BCD is
port(
      A      : in std_logic_vector(3 downto 0);
      B      : in std_logic_vector(3 downto 0);
      Cin    : in std_logic;
      Cout   : buffer std_logic;
      S      : buffer std_logic_vector(3 downto 0)
);
end entity;

architecture rtl of sum_BCD is

      signal sal_sum: std_logic_vector(4 downto 0);  -- Sal sum
      signal sal_rest: std_logic_vector(4 downto 0);

begin

-- Sumador
sal_sum <= ('0'&A) + B + Cin;


-- Restador
sal_rest <= sal_sum - 10;


-- Comparador
Cout <= '1' when sal_sum > 9 else '0';


S <= sal_rest(3 downto 0) when Cout = '1' else sal_sum(3 downto 0);


end rtl;
```

```vhdl
- Test-bench sumador completo BCD de 1 digito


library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity tb_sum_BCD is
end entity;

architecture test of tb_sum_BCD is
```

```
      signal A: std_logic_vector(3 downto 0);
      signal B: std_logic_vector(3 downto 0);
      signal Cout: std_logic;
      signal Cin : std_logic;
      signal S   : std_logic_vector(3 downto 0);


begin
dut: entity Work.sum_BCD(rtl)
port map(
      A => A,
      B => B,
      Cin => Cin,
      Cout => Cout,
      S => S
      );

process
begin


      Cin <= '0';
      B <= (others => '0');
      loop
      A <= (others => '0');



        for i in 0 to 9 loop
            for j in 0 to 9 loop
                      wait for 100 ns;
                      B <= B + 1;
              end loop;
        A <= A + 1;
        B <= (others => '0');
        end loop;

        if Cin = '1' then
           exit;
        end if;
        Cin <= not Cin;
      end loop;


      wait;
end process;

end test;
```