```vhdl
library STD;
library IEEE;
use IEEE.std_logic_1164.all;

entity alu_4 is port(
  A: in std_logic_vector(3 downto 0);
  B: in std_logic_vector(3 downto 0);
  Cin: in std_logic;
  S0: in std_logic;
  S1: in std_logic;
  G: out std_logic_vector(3 downto 0);
  Cout: out std_logic
  );
end alu_4;

architecture structural of alu_4 is
  component all_0 port (
    all_0: out std_logic_vector(3 downto 0));
  end component;
  component all_1 port (
    all_1: out std_logic_vector(3 downto 0));
  end component;
  component inverter port(
    input: in  std_logic;
    output: out std_logic);
  end component;
  component adder4 port(
    input1: in std_logic_vector(3 downto 0);
    input2: in std_logic_vector(3 downto 0);
    carryin: in std_logic;
    sum: out std_logic_vector(3 downto 0);
    carryout: out std_logic);
  end component;
  component mux_4_bit
    port(
      A: in std_logic_vector(3 downto 0);
      B: in std_logic_vector(3 downto 0);
      C: in std_logic_vector(3 downto 0);
      D: in std_logic_vector(3 downto 0);
      SEL0: in std_logic;
      SEL1: in std_logic;
      OUT1: out std_logic_vector(3 downto 0));
  end component;
  for mux: mux_4_bit use entity work.mux_4_bit(structural);
  for adder: adder4 use entity work.adder4(structural);
  for inverter0, inverter1, inverter2, inverter3: inverter use entity work.inverter(structural);

  signal mux_out: std_logic_vector(3 downto 0);
  signal not_B: std_logic_vector(3 downto 0);
  signal all_0_signal: std_logic_vector(3 downto 0):="0000";
  signal all_1_signal: std_logic_vector(3 downto 0):= "1111";
begin
  inverter0: inverter port map(B(0), not_B(0));
  inverter1: inverter port map(B(1), not_B(1));
  inverter2: inverter port map(B(2), not_B(2));
  inverter3: inverter port map(B(3), not_B(3));
  mux: mux_4_bit port map(all_0_signal,B,not_B,all_1_signal,S0,S1,mux_out);
  adder: adder4 port map(A, mux_out,Cin, G, Cout);
end structural;
```