library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.NUMERIC\_STD.ALL;

entity circuit is

    Port ( A : in STD\_LOGIC\_VECTOR (7 downto 0);

           B : in STD\_LOGIC\_VECTOR (7 downto 0);

           op\_cel : in STD\_LOGIC\_VECTOR (3 downto 0);

           c\_in : in STD\_LOGIC\_VECTOR (0 downto 0);

           Disp\_0 : out STD\_LOGIC\_VECTOR (6 downto 0);

           Disp\_1 : out STD\_LOGIC\_VECTOR (6 downto 0));

end circuit;

architecture Behavioral of circuit is

Signal result: STD\_LOGIC\_VECTOR (7 downto 0);

Signal bcd\_0, bcd\_1: std\_logic\_vector(3 downto 0);

begin

    With op\_cel select

result <= std\_logic\_vector(unsigned(A) + unsigned(B)) when "0001",

                 std\_logic\_vector(unsigned(A) - unsigned(B))  when "0010",

                 A AND B  when "0011",

                 A OR B when "0100",

                 std\_logic\_vector(unsigned( NOT B) +1) when "0101",

                 std\_logic\_vector(unsigned( NOT A) + 1) when "0111",

                 std\_logic\_vector(unsigned(A) + 1) when "1000",

                 A NOR B when "1001",

                 std\_logic\_vector(unsigned(A) + unsigned(B) + unsigned(c\_in)) when "1010",

    (others => '0') when others;

bcd\_1 <= result(7) & result(6) & result(5) & result(4);

bcd\_0 <= result(3) & result(2) & result(1) & result(0);

with bcd\_1 select

Disp\_1 <= "1111110" when "0000",

        "0110000" when "0001",

        "1101101" when "0010",

        "1111001" when "0011",

        "0110011" when "0100",

        "1011011" when "0101",

        "1011111" when "0110",

        "1110000" when "0111",

        "1111111" when "1000",

        "1111011" when "1001",

        "1111101" when "1010",

         "0011111" when "1011",

         "1001110" when "1100",

         "0111101" when "1101",

         "1001111" when "1110",

         "1000111" when "1111",

        (others => '0') when others;

        with bcd\_0 select

        Disp\_0 <= "1111110" when "0000",

                "0110000" when "0001",

                "1101101" when "0010",

                "1111001" when "0011",

                "0110011" when "0100",

                "1011011" when "0101",

                "1011111" when "0110",

                "1110000" when "0111",

                "1111111" when "1000",

                "1111011" when "1001",

                "1111101" when "1010",

               "0011111" when "1011",

                "1001110" when "1100",

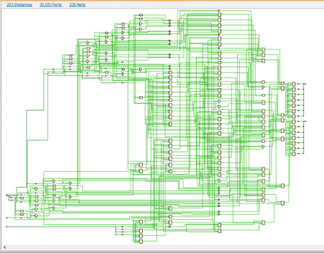
                "0111101" when "1101",

                "1001111" when "1110",

                "1000111" when "1111",

                (others => '0') when others;

end Behavioral;



—------------------------------------------------------------------------------------------------------------------

LIBRARY IEEE;

USE ieee.std\_logic\_1164.ALL;

USE IEEE.numeric\_std.ALL;

ENTITY test IS

END test;

ARCHITECTURE behavior OF test IS

    COMPONENT circuit

    PORT(

         A : IN  std\_logic\_vector(7 downto 0);

         B : IN  std\_logic\_vector(7 downto 0);

         op\_cel : IN  std\_logic\_vector(3 downto 0);

         c\_in : IN  std\_logic\_vector(0 downto 0);

         Disp\_0 : OUT  std\_logic\_vector(6 downto 0);

         Disp\_1 : OUT  std\_logic\_vector(6 downto 0)

        );

    END COMPONENT;

   signal A\_test : std\_logic\_vector(7 downto 0) := (others => '0');

   signal B\_test : std\_logic\_vector(7 downto 0) := (others => '0');

   signal op\_cel\_test : std\_logic\_vector(3 downto 0) := (others => '0');

   signal c\_in\_test : std\_logic\_vector(0 downto 0) := (others => '0');

   signal Disp\_0\_test : std\_logic\_vector(6 downto 0);

   signal Disp\_1\_test : std\_logic\_vector(6 downto 0);

BEGIN

   uut: my\_circ PORT MAP (

          A => A\_test,

          B => B\_test,

          op\_cel => op\_cel\_test,

          c\_in => c\_in\_test,

          Disp\_0 => Disp\_0\_test,

          Disp\_1 => Disp\_1\_test

        );

   process

   begin

A\_test <= "00001010";

B\_test <= "01010010";

c\_in\_test <= "1";

op\_cel\_test <= "0000";

wait for 100 ns;

op\_cel\_test <= "0001";

wait for 100 ns;

op\_cel\_test <= "0010";

wait for 100 ns;

op\_cel\_test <= "0011";

wait for 100 ns;

op\_cel\_test <= "0100";

wait for 100 ns;

op\_cel\_test <= "0101";

wait for 100 ns;

op\_cel\_test <= "0111";

wait for 100 ns;

op\_cel\_test <= "1000";

wait for 100 ns;

op\_cel\_test <= "1001";

wait for 100 ns;

   end process;

END;

Diagram, engineering drawing

Description automatically generated