```vhdl
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
use ieee.numeric_bit.all;
use ieee.numeric_std.all;
use ieee.std_logic_misc.all;

entity octal_7seg is
   port(
      neg : in std_logic;
      octal_in : in std_logic_vector(7 downto 0);
      sign, hexTop, hexDown : out std_logic_vector(6 downto 0));
end octal_7seg;

architecture behavior of octal_7seg is
   signal bcdTop, bcdDown : std_logic_vector(3 downto 0);
      begin
         process(octal_in)
            begin
               bcdTop <= octal_in(7 downto 4);
               bcdDown <= octal_in(3 downto 0);
                  case bcdTop is
                     when "0000" => hexTop <= "0000001"; --0
                     when "0001" => hexTop <= "1001111"; --1
                     when "0010" => hexTop <= "0010010"; --2
                     when "0011" => hexTop <= "0000110"; --3
                     when "0100" => hexTop <= "1001100"; --4
                     when "0101" => hexTop <= "0100100"; --5
                     when "0110" => hexTop <= "0100000"; --6
                     when "0111" => hexTop <= "0001111"; --7
                     when "1000" => hexTop <= "0000000"; --8
                     when "1001" => hexTop <= "0000100"; --9
                     when "1010" => hexTop <= "0001000"; --A
                     when "1011" => hexTop <= "1100000"; --b
                     when "1100" => hexTop <= "0110001"; --C
                     when "1101" => hexTop <= "1000010"; --d
                     when "1110" => hexTop <= "0110000"; --E
                     when "1111" => hexTop <= "0111000"; --F
                     when others => hexTop <= (others => '0');
                  end case;
                  case bcdDown is
                     when "0000" => hexDown <= "0000001"; --0
                     when "0001" => hexDown <= "1001111"; --1
                     when "0010" => hexDown <= "0010010"; --2
                     when "0011" => hexDown <= "0000110"; --3
                     when "0100" => hexDown <= "1001100"; --4
                     when "0101" => hexDown <= "0100100"; --5
                     when "0110" => hexDown <= "0100000"; --6
                     when "0111" => hexDown <= "0001111"; --7
                     when "1000" => hexDown <= "0000000"; --8
                     when "1001" => hexDown <= "0000100"; --9
                     when "1010" => hexDown <= "0001000"; --A
                     when "1011" => hexDown <= "1100000"; --b
                     when "1100" => hexDown <= "0110001"; --C
                     when "1101" => hexDown <= "1000010"; --d
                     when "1110" => hexDown <= "0110000"; --E
                     when "1111" => hexDown <= "0111000"; --F
                     when others => hexDown <= (others => '0');
                  end case;
         if neg = '0' then
            sign <= "1111110";
         elsif neg = '1' then
            sign <= "1111111";
         end if;
      end process;
end behavior;

```