```vhdl
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
use ieee.std_logic_unsigned.all;

package meu_pacote is
   component multidisp is
      port(mclk: in std_logic;
           sd, selout: out std_logic_vector(3 downto 0));
   end component;

   component contador is
      generic(freq, valor: integer);
      port(mclk, reset: in std_logic;
            clk: out std_logic_vector(15 downto 0));
   end component;

   component bcd_decod is
   port(result: in std_logic_vector(15 downto 0);
        selout: in std_logic_vector(3 downto 0);
        saida: out std_logic_vector(6 downto 0));
   end component;

   function to_bcd (bin : std_logic_vector(15 downto 0) ) return std_logic_vector;
end meu_pacote;

package body meu_pacote is
   function to_bcd ( bin : std_logic_vector(15 downto 0) ) return std_logic_vector is
   variable i : integer:=0;
   variable bcd : std_logic_vector(15 downto 0) := (others => '0');
   variable bint : std_logic_vector(15 downto 0) := bin;

   begin
   for i in 0 to 15 loop
      bcd(15 downto 1) := bcd(14 downto 0);
      bcd(0) := bint(15);
      bint(15 downto 1) := bint(14 downto 0);
      bint(0) :='0';

      if(i < 15 and bcd(3 downto 0) > "0100") then
         bcd(3 downto 0) := bcd(3 downto 0) + "0011";
      end if;

      if(i < 15 and bcd(7 downto 4) > "0100") then
         bcd(7 downto 4) := bcd(7 downto 4) + "0011";
      end if;

      if(i < 15 and bcd(11 downto 8) > "0100") then
         bcd(11 downto 8) := bcd(11 downto 8) + "0011";
      end if;

      if(i < 15 and bcd(15 downto 12) > "0100") then
         bcd(15 downto 12) := bcd(15 downto 12) + "0011";
      end if;
   end loop;
   return bcd;
end to_bcd;
end meu_pacote;

```