```
-- LED to BCD code converter
library IEEE;
use IEEE.std_logic_1164.all;
-- I/O pin
entity LED2BCD is
        port(
                CLK     : in    std_logic;                                              -- clock
                SEG7    : in    std_logic_vector(6 downto 0);   -- LED segment (a,b,c,d,e,f,g)
                LED_S   : in    std_logic_vector(2 downto 0);   -- LED select
(.....,011:1k,100:10k,101:100k,110:1M,111:10M)
                F10M    : out   std_logic_vector(3 downto 0);   -- BCD out  10MHz
                F01M    : out   std_logic_vector(3 downto 0);   -- BCD out   1MHz
                F0100k  : out   std_logic_vector(3 downto 0);   -- BCD out 100kHz
                F010k   : out   std_logic_vector(3 downto 0);   -- BCD out  10kHz
                F01k    : out   std_logic_vector(3 downto 0)    -- BCD out   1kHz
                );
end LED2BCD;
--
--
architecture RTL of LED2BCD is
-- internal signals
signal  F10M    : std_logic_vector(3 downto 0);
signal  F1M     : std_logic_vector(3 downto 0);
signal  F100k   : std_logic_vector(3 downto 0);
signal  F10k    : std_logic_vector(3 downto 0);
signal  F1k     : std_logic_vector(3 downto 0);
signal  BCD     : std_logic_vector(3 downto 0);
signal  buff    : std_logic_vector(15 downto 0);
signal  reg1    : std_logic;    -- 1 clock delay from LED_S(0)
signal  reg2    : std_logic;    -- 2 clock delay from LED_S(0)
signal  enable  : std_logic;    -- data transfer to the output registors

begin

-- 7 segment to BCD converter
        BCD <=  "0000" when SEG7 = "1111110" else       -- 0
                        "0001" when SEG7 = "0110000" else       -- 1
                        "0010" when SEG7 = "1101101" else       -- 2
                        "0011" when SEG7 = "1111001" else       -- 3
                        "0100" when SEG7 = "0110011" else       -- 4
                        "0101" when SEG7 = "1011011" else       -- 5
                        "0110" when SEG7 = "1011111" else       -- 6
                        "0111" when SEG7 = "1110000" else       -- 7
                        "1000" when SEG7 = "1111111" else       -- 8
                        "1001" when SEG7 = "1111011" else       -- 9
                        "1010" when SEG7 = "1110111" else       -- A
                        "1011" when SEG7 = "0011111" else       -- B
                        "1100" when SEG7 = "1001110" else       -- C
                        "1101" when SEG7 = "0111101" else       -- D
                        "1110" when SEG7 = "1001111" else       -- E
                        "1111" when SEG7 = "1000111" else       -- F
                        "0000";

-- de-glitch for LED select
        process (CLK) begin
                if(CLK'event and CLK='1') then
                        reg1 <= LED_S(0);
                        reg2 <= reg1;
                end if;
        end process;

        enable  <= reg1 xor reg2;       -- 1 clock piriod from rising and falling edge of LED_S(0)

-- save BCD to the registors
        process (CLK) begin
                if(CLK'event and CLK='1') then
                        if(LED_S = "011" and enable = '1') then
                                F1k <= BCD;
                        else
                                F1k <= F1k;
                        end if;
                end if;
        end process;

        process (CLK) begin
```

```
                if(CLK'event and CLK='1') then
                        if(LED_S = "100" and enable = '1') then
                                F10k <= BCD;
                        else
                                F10k <= F10k;
                        end if;
                end if;
        end process;

        process (CLK) begin
                if(CLK'event and CLK='1') then
                        if(LED_S = "101" and enable = '1') then
                                F100k <= BCD;
                        else
                                F100k <= F100k;
                        end if;
                end if;
        end process;

        process (CLK) begin
                if(CLK'event and CLK='1') then
                        if(LED_S = "110" and enable = '1') then
                                F1M <= BCD;
                        else
                                F1M <= F1M;
                        end if;
                end if;
        end process;

        process (CLK) begin
                if(CLK'event and CLK='1') then
                        if(LED_S = "111" and enable = '1') then
                                F10M <= BCD;
                                buff(15 downto 12) <= F1M;
                                buff(11 downto  8) <= F100k;
                                buff( 7 downto  4) <= F10k;
                                buff( 3 downto  0) <= F1k;
                        else
                                F10M <= F10M;
                                buff <= buff;
                        end if;
                end if;
        end process;

-- BCD output
        F010M   <= F10M;
        F01M    <= buff(15 downto 12);
        F0100k  <= buff(11 downto 8);
        F010k   <= buff(7 downto 4);
        F01k    <= buff(3 downto 0);

end RTL;
```