```
-- CS214 Practical 5 Memo
-- authors: MJ Booysen and RD Beyers
-- updated on 16 April 2015

LIBRARY ieee;
USE ieee.std_logic_1164.all;
use ieee.numeric_std.all;
LIBRARY altera;
use altera.altera_primitives_components.all;


ENTITY DE0 IS
        PORT
        (
                CLOCK_50 : IN STD_LOGIC; -- 50MHz in-circuit clock
                LEDG : OUT STD_LOGIC_VECTOR(9 DOWNTO 0); -- the 10 green LEDs on
the DE0 board
                SW : IN STD_LOGIC_VECTOR(9 DOWNTO 0); -- the 10 switches on the
DE0 board
                BUTTON : IN STD_LOGIC_VECTOR(0 TO 2);  -- the 3 buttons on the
DE0 board
                HEX0_D : INOUT STD_LOGIC_VECTOR(0 TO 6); -- the LEDs of the
7-segment display (right)
                HEX1_D : INOUT STD_LOGIC_VECTOR(0 TO 6); -- the LEDs of the
7-segment display
                HEX2_D : INOUT STD_LOGIC_VECTOR(0 TO 6); -- the LEDs of the
7-segment display
                HEX3_D : INOUT STD_LOGIC_VECTOR(0 TO 6); -- the LEDs of the
7-segment display (left)
                FL_BYTE_N : IN STD_LOGIC;
                FL_CE_N : IN STD_LOGIC;
                FL_OE_N : IN STD_LOGIC;
                FL_RST_N : IN STD_LOGIC;
                FL_RY : IN STD_LOGIC;
                FL_WE_N : IN STD_LOGIC;
                FL_WP_N : IN STD_LOGIC;
                FL_DQ15_AM1 : IN STD_LOGIC;
                PS2_KBCLK : IN STD_LOGIC;
                PS2_KBDAT : IN STD_LOGIC;
                PS2_MSCLK : IN STD_LOGIC;
                PS2_MSDAT : IN STD_LOGIC;
                UART_RXD : IN STD_LOGIC;
                UART_TXD : IN STD_LOGIC;
                UART_RTS : IN STD_LOGIC;
                UART_CTS : IN STD_LOGIC;
                SD_CLK : IN STD_LOGIC;
                SD_CMD : IN STD_LOGIC;
                SD_DAT0 : IN STD_LOGIC;
                SD_DAT3 : IN STD_LOGIC;
                SD_WP_N : IN STD_LOGIC;
                LCD_RW : IN STD_LOGIC;
                LCD_RS : IN STD_LOGIC;
                LCD_EN : IN STD_LOGIC;
                LCD_BLON : IN STD_LOGIC;
                VGA_HS : IN STD_LOGIC;
                VGA_VS : IN STD_LOGIC;
                HEX0_DP : IN STD_LOGIC;
                HEX1_DP : IN STD_LOGIC;
                HEX2_DP : IN STD_LOGIC;
                HEX3_DP : IN STD_LOGIC;
                DRAM_CAS_N : IN STD_LOGIC;
                DRAM_CS_N : IN STD_LOGIC;
                DRAM_CLK : IN STD_LOGIC;
                DRAM_CKE : IN STD_LOGIC;
                DRAM_BA_0 : IN STD_LOGIC;
                DRAM_BA_1 : IN STD_LOGIC;
                DRAM_LDQM : IN STD_LOGIC;
                DRAM_UDQM : IN STD_LOGIC;
```

```
                DRAM_RAS_N : IN STD_LOGIC;
                DRAM_WE_N : IN STD_LOGIC;
                CLOCK_50_2 : IN STD_LOGIC;
                FL_ADDR : IN STD_LOGIC_VECTOR(0 TO 21);
                FL_DQ : IN STD_LOGIC_VECTOR(0 TO 14);
                GPIO0_D : INOUT STD_LOGIC_VECTOR(0 TO 31);
                GPIO0_CLKIN : IN STD_LOGIC_VECTOR(0 TO 1);
                GPIO0_CLKOUT : IN STD_LOGIC_VECTOR(0 TO 1);
                GPIO1_CLKIN : IN STD_LOGIC_VECTOR(0 TO 1);
                GPIO1_CLKOUT : IN STD_LOGIC_VECTOR(0 TO 1);
                GPIO1_D : IN STD_LOGIC_VECTOR(0 TO 31);
                LCD_DATA : IN STD_LOGIC_VECTOR(0 TO 7);
                VGA_G : IN STD_LOGIC_VECTOR(0 TO 3);
                VGA_R : IN STD_LOGIC_VECTOR(0 TO 3);
                VGA_B : IN STD_LOGIC_VECTOR(0 TO 3);
                DRAM_DQ : IN STD_LOGIC_VECTOR(0 TO 15);
                DRAM_ADDR : IN STD_LOGIC_VECTOR(0 TO 12)
                );
END DE0;

ARCHITECTURE structure OF DE0 IS
        SIGNAL SUB_CLK : STD_LOGIC;
        SIGNAL Counter : INTEGER := 0;
BEGIN
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
--      -- Question 1
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        LEDG(0) <= NOT button(0);
--
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
--      -- Question 2
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        LEDG(1) <= button(1)XOR button(0);
--
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
--      -- Question 3
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        u1: dff PORT MAP (d => NOT Button (0), clk => NOT Button (1),
                                clrn => Button (2), prn=> '1', q => LEDG(2));
--
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
--      -- Question 4
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        WITH SW(3 DOWNTO 0) SELECT
                HEX0_D <= NOT "1111110" WHEN "0000",
                          NOT "0110000" WHEN "0001",
                       "0010010" when "0010",
                           "0000110" when "0011",
                           "1001100" when "0100",
                           "0100100" when "0101",
                           "0100000" when "0110",
                           "0001111" when "0111",
                           "0000000" when "1000",
                           "0000100" when "1001",
                           "0001000" when "1010",
                           "1100000" when "1011",
                           "0110001" when "1100",
                           "1000010" when "1101",
                           "0110000" when "1110",
                           "0111000" when "1111";

--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
--      -- Question 5
--      -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        PROCESS (CLOCK_50)
          VARIABLE SUB_CLK_CNT  : INTEGER RANGE 0 TO 25000000;
        BEGIN
          IF CLOCK_50'EVENT AND CLOCK_50 = '1' THEN
```

```
            SUB_CLK_CNT := SUB_CLK_CNT + 1;
            IF SUB_CLK_CNT = 25000000 THEN
                  SUB_CLK_CNT := 0;
               SUB_CLK <= NOT SUB_CLK;
            END IF;
          END IF;
        END PROCESS;
    LEDG(3)<=    SUB_CLK;

        -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        -- Question 6
        -- xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx --
        PROCESS (SUB_CLK)
        BEGIN
                IF SUB_CLK'EVENT AND SUB_CLK = '1' THEN
                        IF SW(9) = '0' THEN
                                Counter <= Counter + 1;
                                IF Counter >= 99 THEN Counter <= 0; END IF;
                        ELSE
                                Counter <= Counter - 1;
                                IF Counter <= 0 THEN Counter <= 99; END IF;
                        END IF;
                END IF;
        END PROCESS;
        WITH Counter REM 10 SELECT
                HEX2_D <= NOT "1111110" WHEN 0,
                   NOT "0110000" WHEN 1,
                  "0010010" when 2,
                  "0000110" when 3,
                  "1001100" when 4,
                  "0100100" when 5,
                  "0100000" when 6,
                  "0001111" when 7,
                  "0000000" when 8,
                  "0000100" when 9,
                        "0110000" when OTHERS;
        WITH (Counter/10) SELECT
                HEX3_D <= NOT "1111110" WHEN 0,
                  NOT "0110000" WHEN 1,
                  "0010010" when 2,
                  "0000110" when 3,
                  "1001100" when 4,
                  "0100100" when 5,
                  "0100000" when 6,
                  "0001111" when 7,
                  "0000000" when 8,
                  "0000100" when 9,
                "0110000" when OTHERS;

END structure;
```