گزارش کار تمرین شماره ۳

پارسا دارسرایی ۱۴۰۲۰۱۲۲۶۸۱۵۶

بخش الف)

کد

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity Blanking_Circuit is
   Port ( clk : in  STD_LOGIC;
          led_out : out  STD_LOGIC);
end Blanking_Circuit;

architecture Behavioral of Blanking_Circuit is
   signal toggle : STD_LOGIC := '0';
begin
   process(clk)
   begin
      if rising_edge(clk) then
         toggle <= not toggle;
      end if;
   end process;

   led_out <= toggle;
end Behavioral;
```

شبیه سازی با ISim

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity tb_Blanking is
end tb_Blanking;

architecture Behavioral of tb_Blanking is
   signal clk_tb : STD_LOGIC := '0';
   signal led_tb : STD_LOGIC;

   component Blanking_Circuit
      Port ( clk : in  STD_LOGIC;
             led_out : out  STD_LOGIC);
   end component;
begin
   UUT: Blanking_Circuit port map (clk_tb, led_tb);

   clk_tb <= not clk_tb after 10 ns;
end Behavioral;
```

نتیجه شبیه‌سازی:

- سیگنال led_out با هر لبه بالارونده کلاک تغییر حالت می‌دهد
- دوره تناوب خروجی: 20 نانوثانیه
- شکل موج مطابق انتظار تولید شد

تعریف pin ها

NET "clk" LOC = P94; # پایه کلاک اصلی

LED1 پایه # ;NET "led_out" LOC = P125

بخش ب)

کد

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
use IEEE.NUMERIC_STD.ALL;

entity BCD_Down_Counter is
   Port ( clk : in  STD_LOGIC;
          bcd_out : out  STD_LOGIC_VECTOR (15 downto 0));
end BCD_Down_Counter;

architecture Behavioral of BCD_Down_Counter is
   signal units : STD_LOGIC_VECTOR(3 downto 0) := "1001";
   signal tens : STD_LOGIC_VECTOR(3 downto 0) := "1001";
   signal hundreds : STD_LOGIC_VECTOR(3 downto 0) := "1001";
   signal thousands : STD_LOGIC_VECTOR(3 downto 0) := "1001";

begin
   process(clk)
```

```
begin
   if rising_edge(clk) then
      if units = "0000" then
         units <= "1001";
         if tens = "0000" then
            tens <= "1001";
            if hundreds = "0000" then
               hundreds <= "1001";
               if thousands = "0000" then
                  thousands <= "1001";
               else
                  thousands <= thousands - 1;
               end if;
            else
               hundreds <= hundreds - 1;
            end if;
         else
            tens <= tens - 1;
         end if;
      else
         units <= units - 1;
      end if;
   end if;
end process;

bcd_out <= thousands & hundreds & tens & units;
```

```
end Behavioral;
```

شبیه سازی با ISim

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity tb_BCD_Counter is
end tb_BCD_Counter;

architecture Behavioral of tb_BCD_Counter is
   signal clk_tb : STD_LOGIC := '0';
   signal bcd_out_tb : STD_LOGIC_VECTOR(15 downto 0);

   component BCD_Down_Counter
      Port ( clk : in  STD_LOGIC;
            bcd_out : out  STD_LOGIC_VECTOR (15 downto 0));
   end component;
begin
   UUT: BCD_Down_Counter port map (clk_tb, bcd_out_tb);

   clk_tb <= not clk_tb after 10 ns;
end Behavioral;
```

نتیجه شبیه‌سازی:

- شمارنده از 9999 شروع به شمارش نزولی می‌کند
- پس از رسیدن به 0000، مجدداً به 9999 بازمی‌گردد
- هر رقم BCD به درستی مدیریت می‌شود

```
NET "clk" LOC = P94;

NET "bcd_out[0]" LOC = P125;

NET "bcd_out[1]" LOC = P126;

...

NET "bcd_out[15]" LOC = P140;
```

بخش الف:

- مدار طراحی شده به درستی با سیگنال کلاک همگام شده است
- خروجی LED با فرکانس نصف فرکانس کلاک تغییر می‌کند
- شکل موج خروجی کاملاً مطابق با انتظارات طراحی است

بخش ب:

- الگوریتم کاهش BCD به درستی پیاده‌سازی شده است
- مدیریت انتقال بین ارقام (units → tens → hundreds → thousands) صحیح کار می‌کند
- بازگشت خودکار از 0000 به 9999 به درستی انجام می‌شود
- شمارنده در شبیه‌سازی عملکرد پایدار و قابل پیش‌بینی دارد