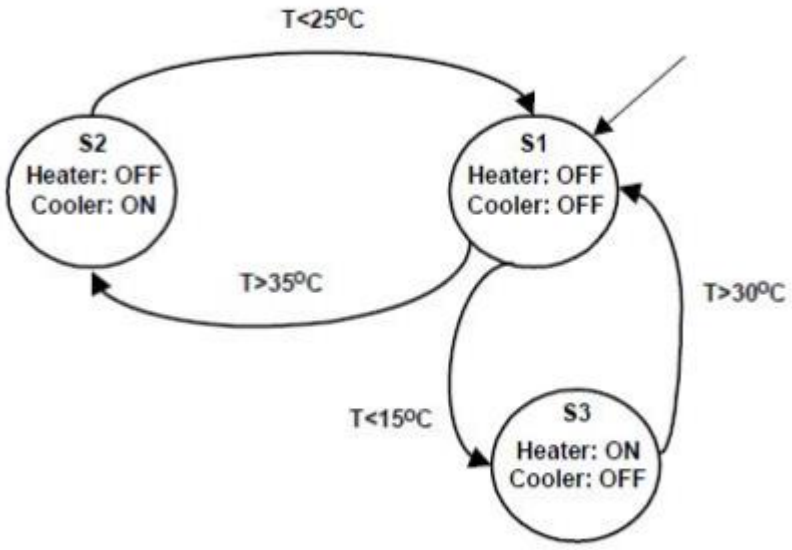
به نام خدا

حسین عبدالهی پور

ابتدا یک entity برای قسمتی می نویسیم که کنترل می کند کولر روشن شود یا بخاری و یا هیچ کدام. نام این entity ، heat_cooler می باشد که کد آن و شکل آن را داریم:




```
----------------------------------------------------------------------------------

-- Company:

-- Engineer:

--

-- Create Date: 08/31/2020 11:21:40 AM

-- Design Name:
```

```
-- Module Name: heat_cooler - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
```

```
--library UNISIM;
--use UNISIM.VComponents.all;

entity heat_cooler is
   generic ( number: integer := 8);
   Port (
      temp: in std_logic_vector(number-1 downto 0);
      clk, reset: in std_logic;
      cooler_on, heater_on: out std_logic
      );
end heat_cooler;

architecture Behavioral of heat_cooler is

   subtype STATE_TYPE is std_ulogic_vector(1 downto 0);
   constant S1: STATE_TYPE:= "00";
   constant S2: STATE_TYPE:= "01";
   constant S3: STATE_TYPE:= "11";

   signal cooler_on_I, heater_on_I: std_logic;
   signal STATE, NEXTSTATE: STATE_TYPE;

begin
```

```
REG: process(clk, reset)
begin
   if reset = '1' then
      STATE <= S1;
   elsif clk'event and clk='1' then
      STATE <= NEXTSTATE;
   end if;
end process REG;


CMB: process(STATE, temp)
begin
   case STATE is
      when S1 =>
         if temp > "00100011" and temp < "10000000" then
            NEXTSTATE <= S2;
         elsif temp < "00001111" then
            NEXTSTATE <= S3;
         elsif temp > "01111111" then
            NEXTSTATE <= S3;
         elsif temp > "00001110" and temp < "00100100" then
            nextstate <= s1;
         end if;
      when S2 =>
```

```
        if temp < "00011001" then
           NEXTSTATE <= S1;
        elsif temp > "00011000" and temp < "10000000" then
           NEXTSTATE <= S2;
        elsif temp > "01111111" then
           NEXTSTATE <= S1;
        end if;
      when S3 =>
        if temp > "00011110" and temp < "10000000" then
           NEXTSTATE <= S1;
        elsif temp < "00011111" then
           NEXTSTATE <= S3;
        elsif temp > "01111111" then
           NEXTSTATE <= S3;
        end if;
      when others =>
        NEXTSTATE <= S1;
   end case;
end process CMB;


OUTPUT: process(NEXTSTATE, temp)
begin
   case NEXTSTATE is
```

```
        when S1 =>
           cooler_on_I <= '0';
           heater_on_I <= '0';
        when S2 =>
           cooler_on_I <= '1';
           heater_on_I <= '0';
        when S3 =>
           cooler_on_I <= '0';
           heater_on_I <= '1';
        when others =>
           cooler_on_I <= '0';
           heater_on_I <= '0';
    end case;
end process OUTPUT;


-- clocked output process
OUTPUT_REG: process(clk)
begin
    if clk'event and clk='1' then
        cooler_on <= cooler_on_I;
        heater_on <= heater_on_I;
    end if;
end process OUTPUT_REG;
```

end Behavioral;

این قسمت، کنترل روشن یا خاموش شدن کولر و بخاری را در اختیار دارد. حال برای این قسمت testbench نوشته و داریم:

```
----------------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 08/31/2020 12:03:16 PM
-- Design Name:
-- Module Name: tb_heat_cooler - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
```

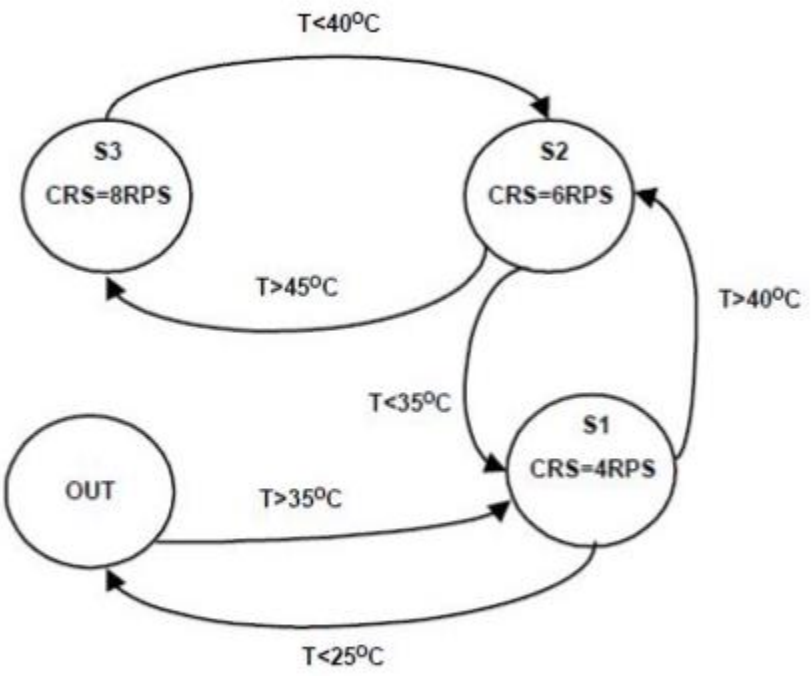
```
--------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity tb_heat_cooler is
end tb_heat_cooler;

architecture Behavioral of tb_heat_cooler is

   constant number: integer := 8;
   signal temp_I: std_logic_vector(number-1 downto 0);
   signal clk_I, reset_I, cooler_on_I, heater_on_I: std_logic;
```

```
begin

    UUT: entity work.heat_cooler
    port map(
        temp => temp_I,
        clk => clk_I,
        reset => reset_I,
        cooler_on => cooler_on_I,
        heater_on => heater_on_I
        );



    tf: process
    begin

        for i in 0 to 20 loop
            clk_I <= '0';
            wait for 30ns;
            clk_I <= '1';
            wait for 30ns;
        end loop;

    end process tf;
```

```
    tb: process
    begin

        reset_I <= '1';
        wait for 1ps;
        reset_I <= '0';
        wait for 1ps;

        temp_I <= "11111011";
        wait for 500ns;

        temp_I <= "00100100";
        wait for 500ns;

--      temp_I <= "11111011";
        temp_I <= "00001010";
        wait for 500ns;

        temp_I <= "00110111";
        wait for 500ns;

        temp_I <= "00011000";
        wait for 500ns;
```

```
    end process tb;

end Behavioral;
```

حال با توجه به شکل زیر:



CRS: Cooler Rotational Speed

اگر کد این قسمت را بنویسیم داریم:

```
----------------------------------------------------------------------------------

-- Company:

-- Engineer:
```

```
--
-- Create Date: 08/30/2020 01:48:03 PM
-- Design Name:
-- Module Name: cooler - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;
```

```
-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity cooler is
   generic ( number: integer := 8);
   Port (
      temp: in std_logic_vector(number-1 downto 0);
      clk, cooler_on: in std_logic;
      speed: out std_logic_vector(3 downto 0)
      );
end cooler;

architecture Behavioral of cooler is

   subtype STATE_TYPE is std_ulogic_vector(1 downto 0);
   constant S1: STATE_TYPE:= "00";
   constant S2: STATE_TYPE:= "01";
   constant S3: STATE_TYPE:= "11";
   constant S_out: STATE_TYPE:= "10";

   signal speed_I: std_logic_vector(3 downto 0);
```

```
    signal STATE, NEXTSTATE: STATE_TYPE := "00";

begin

    REG: process(clk, cooler_on)
    begin
        if cooler_on = '0' then
            STATE <= S_out;
        elsif clk'event and clk='1' then
            STATE <= NEXTSTATE;
        end if;
    end process REG;


    CMB: process(STATE, temp)
    begin
        case STATE is
            when S_out =>
                if temp > "00100011" and temp < "10000000" then
                    NEXTSTATE <= S1;
                elsif temp < "00100100" then
                    NEXTSTATE <= S_out;
                elsif temp > "01111111" then
                    NEXTSTATE <= S_out;
```

```
    end if;
when S1 =>
    if temp > "00101000" and temp < "10000000" then
        NEXTSTATE <= S2;
    elsif temp < "00011001" then
        NEXTSTATE <= S_out;
    elsif temp > "01111111" then
        NEXTSTATE <= S_out;
    elsif temp < "00101001" and temp > "00011000" then
        NEXTSTATE <= S1;
    end if;
when S2 =>
    if temp < "00100011" then
        NEXTSTATE <= S1;
    elsif temp > "01111111" then
        NEXTSTATE <= S1;
    elsif temp > "00101101" and temp < "10000000" then
        NEXTSTATE <= S3;
    elsif temp > "00100010" and temp < "00101110" then
        NEXTSTATE <= S2;
    end if;
when S3 =>
    if temp < "00101000" then
        NEXTSTATE <= S2;
```

```
        elsif temp > "01111111" then
            NEXTSTATE <= S2;
        elsif temp > "00100111" and temp < "10000000" then
            NEXTSTATE <= S3;
        end if;
      when others =>
        NEXTSTATE <= S_out;
   end case;
end process CMB;


OUTPUT: process(NEXTSTATE, temp)
begin
   case NEXTSTATE is
      when S_out =>
        speed_I <= "0000";
      when S1 =>
        speed_I <= "0100";
      when S2 =>
        speed_I <= "0110";
      when S3 =>
        speed_I <= "1000";
      when others =>
        speed_I <= "0000";
```

```
    end case;
  end process OUTPUT;



  -- clocked output process
  OUTPUT_REG: process(clk)
  begin
    if clk'event and clk='1' then
      speed <= speed_I;
    end if;
  end process OUTPUT_REG;



end Behavioral;
```

و اگر برای این قسمت از کد نیز testbench بنویسیم، داریم:

```
----------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 08/31/2020 09:33:40 AM
-- Design Name:
```

```
-- Module Name: tb_cooler - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
```

```
--library UNISIM;
--use UNISIM.VComponents.all;

entity tb_cooler is
end tb_cooler;

architecture Behavioral of tb_cooler is

    constant number: integer := 8;
    signal temp_I: std_logic_vector(number-1 downto 0);
    signal clk_I, cooler_on_I: std_logic;
    signal speed_I: std_logic_vector(3 downto 0);

begin

    cooler_on_I <= '1';

    UUT: entity work.cooler
    port map(
        temp => temp_I,
        clk => clk_I,
        cooler_on => cooler_on_I,
        speed => speed_I
        );
```

```
tf: process
begin

   for i in 0 to 20 loop
      clk_I <= '0';
      wait for 30ns;
      clk_I <= '1';
      wait for 30ns;
   end loop;

end process tf;


tb: process
begin

   temp_I <= "11111011";
   wait for 500ns;

   temp_I <= "00100100";
   wait for 500ns;
```

```
--       temp_I <= "11111011";
        temp_I <= "00001010";
        wait for 500ns;

        temp_I <= "00110111";
        wait for 500ns;

        temp_I <= "00011000";
        wait for 500ns;

    end process tb;

end Behavioral;
```

در نهایت تمامی کد ها را تحت یک قسمت با نام control تعریف می کنیم و داریم:

```
----------------------------------------------------------------------------
-- Company:
-- Engineer:
--
-- Create Date: 08/31/2020 11:12:01 AM
-- Design Name:
-- Module Name: control - Behavioral
```

```
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
```

```
--use UNISIM.VComponents.all;

entity control is
   generic ( number: integer := 8);
   Port (
      temp: in std_logic_vector(number-1 downto 0);
      clk, reset: in std_logic;
      cooler_LED, heater_LED: out std_logic;
      speed: out std_logic_vector(3 downto 0)
      );
end control;

architecture Behavioral of control is

   component cooler is
      generic ( number: integer := 8);
      Port (
         temp: in std_logic_vector(number-1 downto 0);
         clk, cooler_on: in std_logic;
         speed: out std_logic_vector(3 downto 0)
         );
   end component;

   for all: cooler use entity work.cooler(Behavioral);
```

```
    component heat_cooler is
        generic ( number: integer := 8);
        Port (
            temp: in std_logic_vector(number-1 downto 0);
            clk, reset: in std_logic;
            cooler_on, heater_on: out std_logic
            );
    end component;

    for all: heat_cooler use entity work.heat_cooler(Behavioral);



--    signal temp_I: std_logic_vector(number-1 downto 0) := "00000000";
    signal cooler_LED_I: std_logic := '0';

begin

    cooler_config: cooler
    port map (
        temp => temp,
        clk => clk,
        cooler_on => cooler_LED_I,
```

```
        speed => speed
        );

    heat_cooler_config: heat_cooler
    port map (
        temp => temp,
        clk => clk,
        reset => reset,
        cooler_on => cooler_LED_I,
        heater_on => heater_LED
        );

    cooler_LED <= cooler_LED_I;

--  porting: process
--  begin

--  end process porting;

end Behavioral;
```

حال برای این قسمت از کد که کد اصلی پروژه نیز می باشد، testbench می نویسیم و داریم:

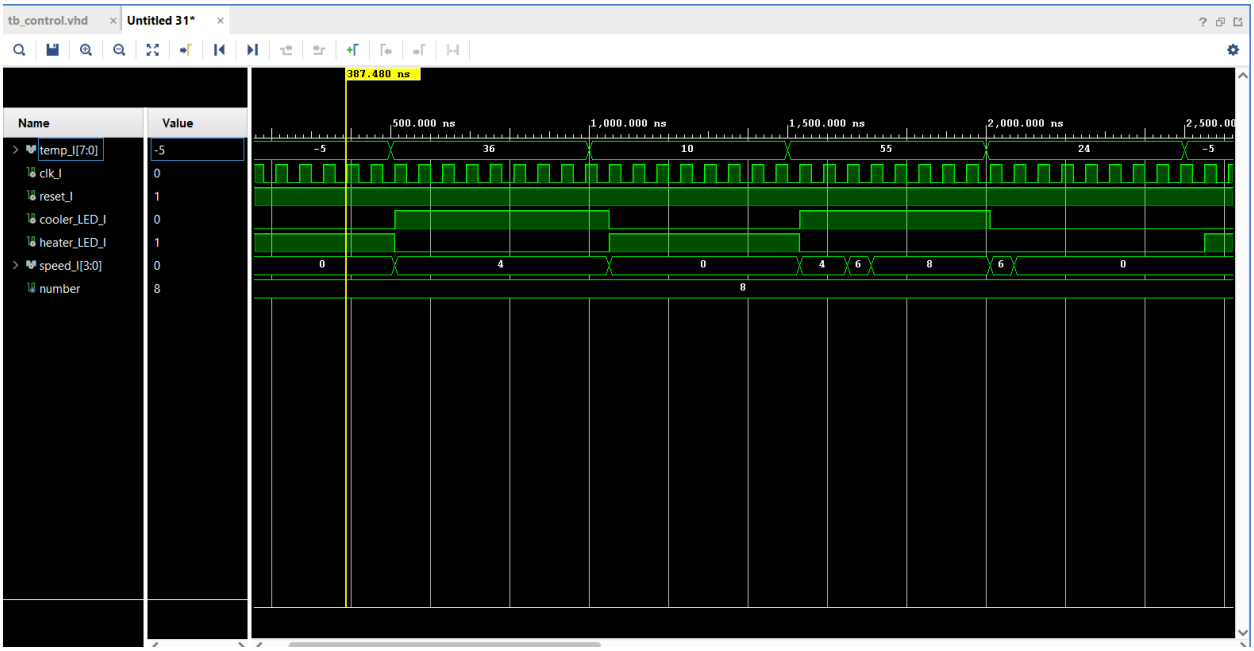
```
-- Company:
-- Engineer:
--
-- Create Date: 08/31/2020 09:40:08 PM
-- Design Name:
-- Module Name: tb_control - Behavioral
-- Project Name:
-- Target Devices:
-- Tool Versions:
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
----------------------------------------------------------------------------------


library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

-- Uncomment the following library declaration if using
```

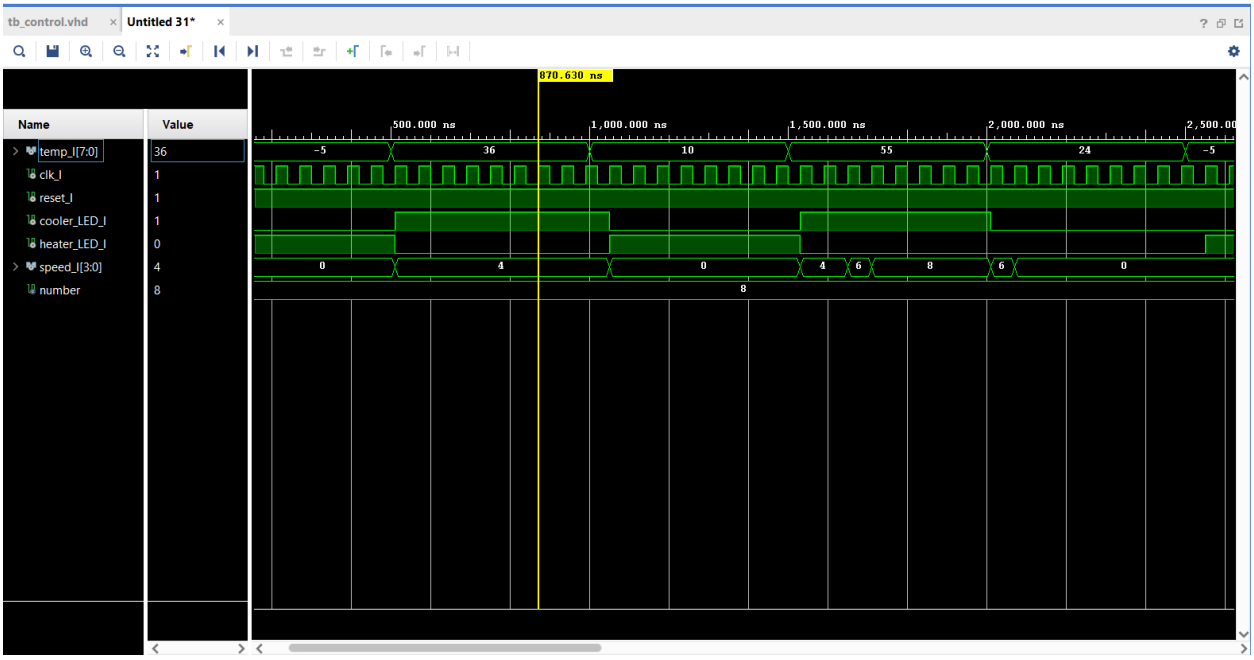
```
-- arithmetic functions with Signed or Unsigned values
--use IEEE.NUMERIC_STD.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx leaf cells in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity tb_control is
end tb_control;

architecture Behavioral of tb_control is

   constant number: integer := 8;
   signal temp_I: std_logic_vector(number-1 downto 0);
   signal clk_I, reset_I, cooler_LED_I, heater_LED_I: std_logic ;
   signal speed_I: std_logic_vector(3 downto 0);

begin

   UUT: entity work.control
   port map(
      temp => temp_I,
      clk => clk_I,
```

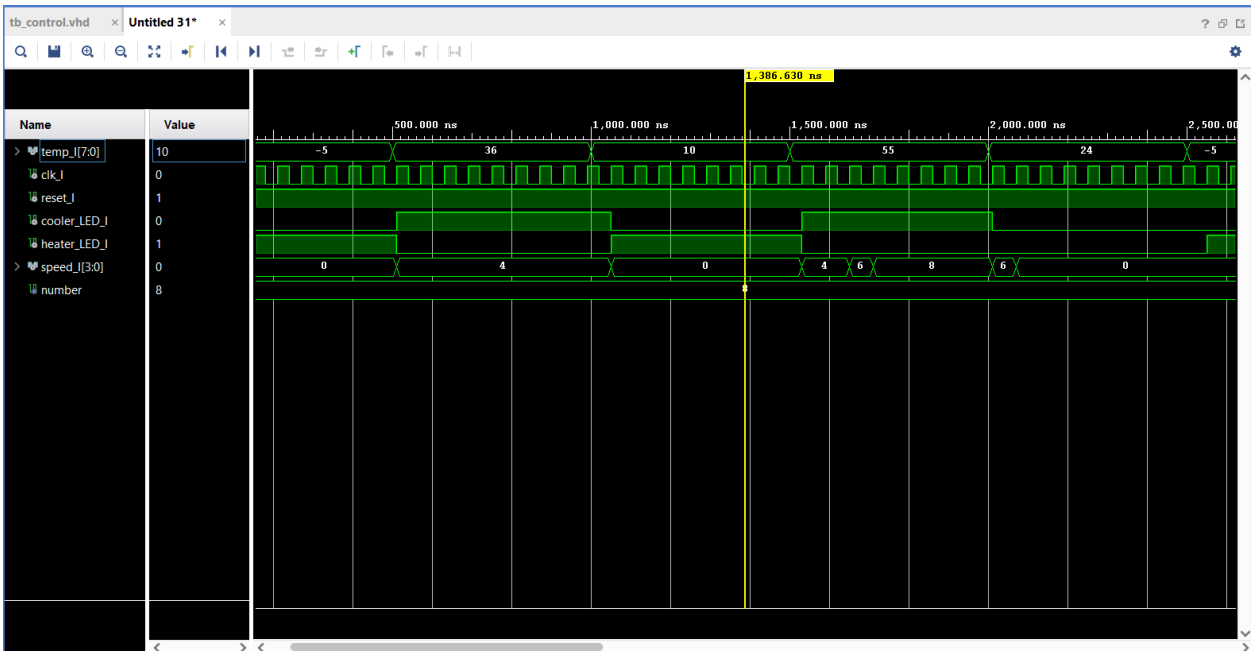
```
    reset => reset_I,
    cooler_LED => cooler_LED_I,
    heater_LED => heater_LED_I,
    speed => speed_I
    );


tf: process
begin

    reset_I <= '1';

    for i in 0 to 20 loop
        clk_I <= '0';
        wait for 30ns;
        clk_I <= '1';
        wait for 30ns;
    end loop;

end process tf;

tb: process
begin
```

```
        temp_I <= "11111011";
        wait for 500ns;

        temp_I <= "00100100";
        wait for 500ns;

        temp_I <= "00001010";
        wait for 500ns;

        temp_I <= "00110111";
        wait for 500ns;

        temp_I <= "00011000";
        wait for 500ns;


    end process tb;

end Behavioral;
```

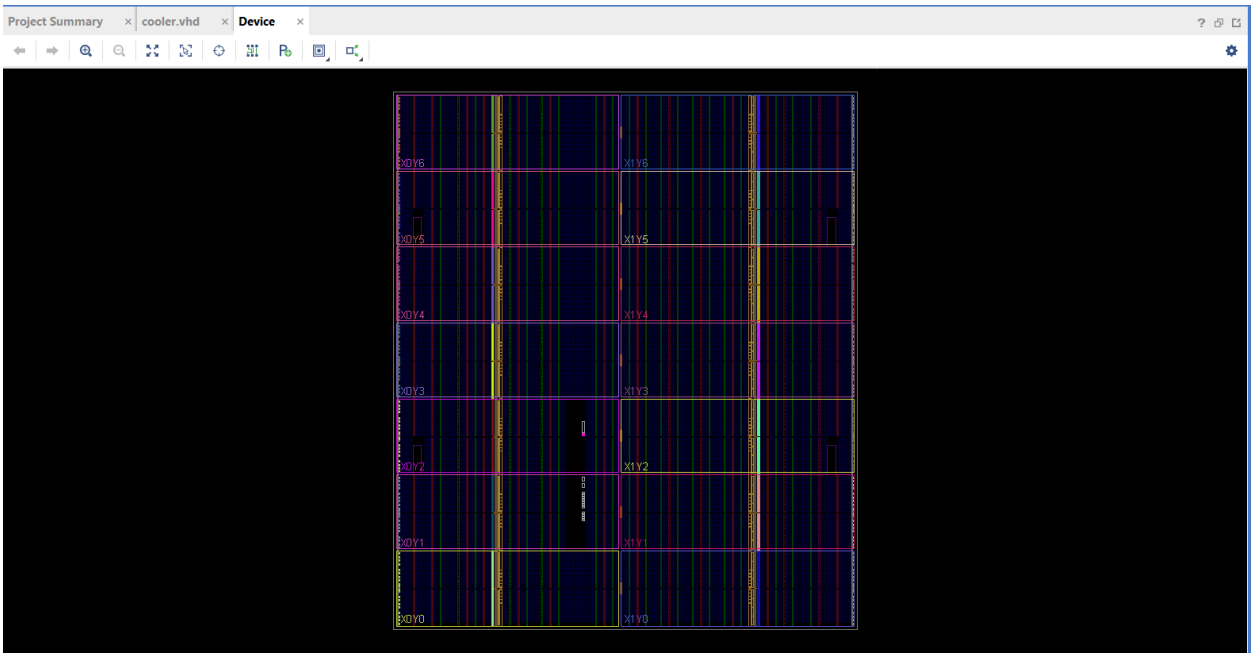
حال این testbench را شبیه سازی می کنیم و در چند مورد، تصاویر آن را مشاهده می کنیم:


tb_control.vhd
Untitled 31*
387.480 ns
Name
Value
temp_I[7:0]
-5
clk_I
0
reset_I
1
cooler_LED_I
0
heater_LED_I
1
speed_I[3:0]
0
number
8
500.000 ns
1,000.000 ns
1,500.000 ns
2,000.000 ns
-5
36
10
55
24
-5
0
4
0
4
6
8
6
0
8


tb_control.vhd
Untitled 31*
870.630 ns
Name
Value
temp_I[7:0]
36
clk_I
1
reset_I
1
cooler_LED_I
1
heater_LED_I
0
speed_I[3:0]
4
number
8
500.000 ns
1,000.000 ns
1,500.000 ns
2,000.000 ns
-5
36
10
55
24
-5
0
4
0
4
6
8
6
0
8

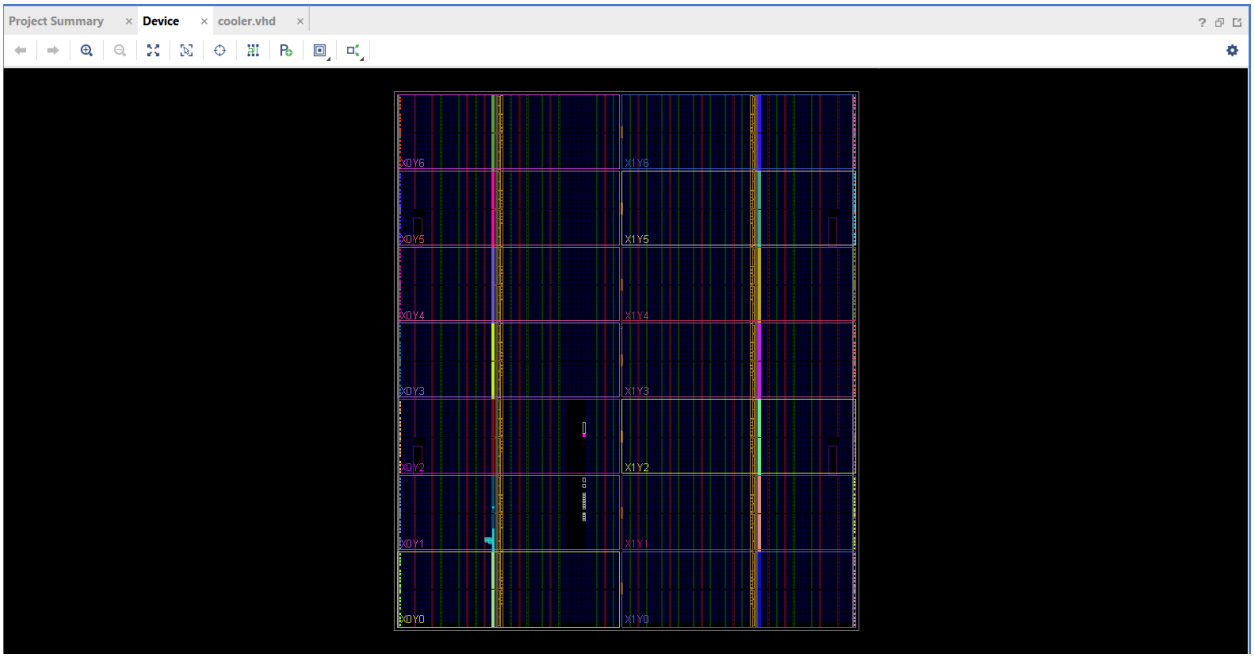


حال این کد را سنتز می کنیم و داریم:



همچنین گزارش های مربوط به قسمت سنتز در فایل ارسالی، با پوشه ای به نام « سنتز »، پیوست شده اند و از آن قسمت قابل مشاهده خواهند بود.

همچنین پس از سنتز کردن، آن را پیاده سازی می کنیم و داریم:



برای این قسمت نیز گزارش هایی به دست می آید که آن ها در پوشه ای به نام « پیاده_سازی » قرار داده شده اند.