

## תרגיל מעבדה – מונה מ-0 עד 9 כל שניה עם תצוגת 7segment

כרטיס DE210-Lite

הדקים והרכיב לניסוי

כניסת שעון לפי האיור הבא:

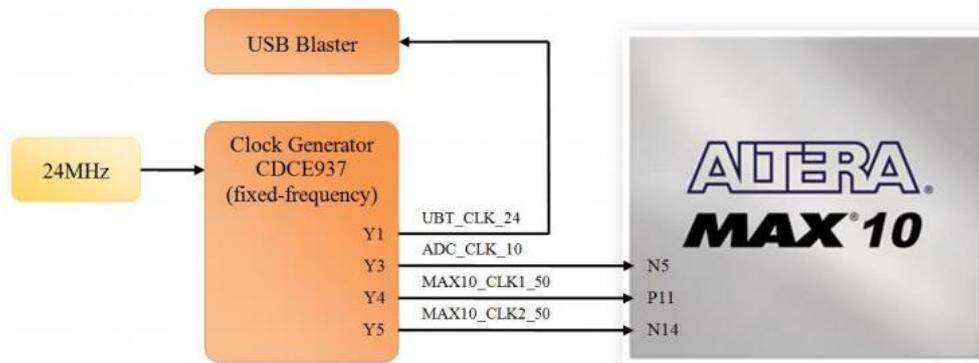


Figure 3-12 Clock circuit of the FPGA Board

Table 3-2 Pin Assignment of Clock Inputs

Signal Name	FPGA Pin No.	Description	I/O Standard
ADC_CLK_10	PIN_N5	10 MHz clock input for ADC (Bank 3B)	3.3-V LVTTTL
MAX10_CLK1_50	PIN_P11	50 MHz clock input(Bank 3B)	3.3-V LVTTTL
MAX10_CLK2_50	PIN_N14	50 MHz clock input(Bank 3B)	3.3-V LVTTTL

- בחירת הפינים לחיבור לדים, מפסקים, לחצנים, 7segment לפי הטבלאות הבאות:

Table 3-3 Pin Assignment of Push-buttons

Signal Name	FPGA Pin No.	Description	I/O Standard
KEY0	PIN_B8	Push-button[0]	3.3 V SCHMITT TRIGGER"
KEY1	PIN_A7	Push-button[1]	3.3 V SCHMITT TRIGGER"

Table 3-4 Pin Assignment of Slide Switches

Signal Name	FPGA Pin No.	Description	I/O Standard
SW0	PIN_C10	Slide Switch[0]	3.3-V LVTTTL
SW1	PIN_C11	Slide Switch[1]	3.3-V LVTTTL
SW2	PIN_D12	Slide Switch[2]	3.3-V LVTTTL
SW3	PIN_C12	Slide Switch[3]	3.3-V LVTTTL
SW4	PIN_A12	Slide Switch[4]	3.3-V LVTTTL
SW5	PIN_B12	Slide Switch[5]	3.3-V LVTTTL
SW6	PIN_A13	Slide Switch[6]	3.3-V LVTTTL
SW7	PIN_A14	Slide Switch[7]	3.3-V LVTTTL
SW8	PIN_B14	Slide Switch[8]	3.3-V LVTTTL
SW9	PIN_F15	Slide Switch[9]	3.3-V LVTTTL

Table 3-5 Pin Assignment of LEDs

Signal Name	FPGA Pin No.	Description	I/O Standard
LEDR0	PIN_A8	LED [0]	3.3-V LVTTTL
LEDR1	PIN_A9	LED [1]	3.3-V LVTTTL
LEDR2	PIN_A10	LED [2]	3.3-V LVTTTL
LEDR3	PIN_B10	LED [3]	3.3-V LVTTTL
LEDR4	PIN_D13	LED [4]	3.3-V LVTTTL
LEDR5	PIN_C13	LED [5]	3.3-V LVTTTL
LEDR6	PIN_E14	LED [6]	3.3-V LVTTTL
LEDR7	PIN_D14	LED [7]	3.3-V LVTTTL
LEDR8	PIN_A11	LED [8]	3.3-V LVTTTL
LEDR9	PIN_B11	LED [9]	3.3-V LVTTTL

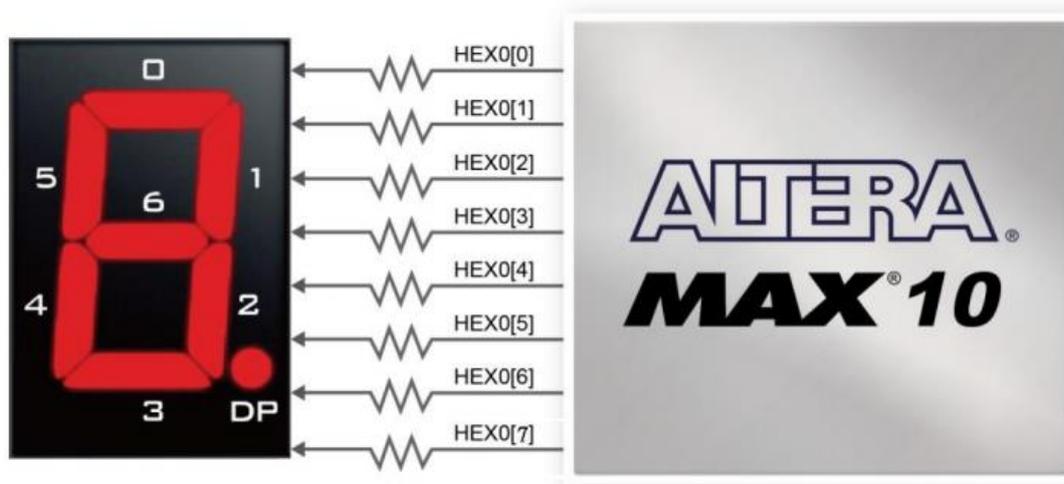


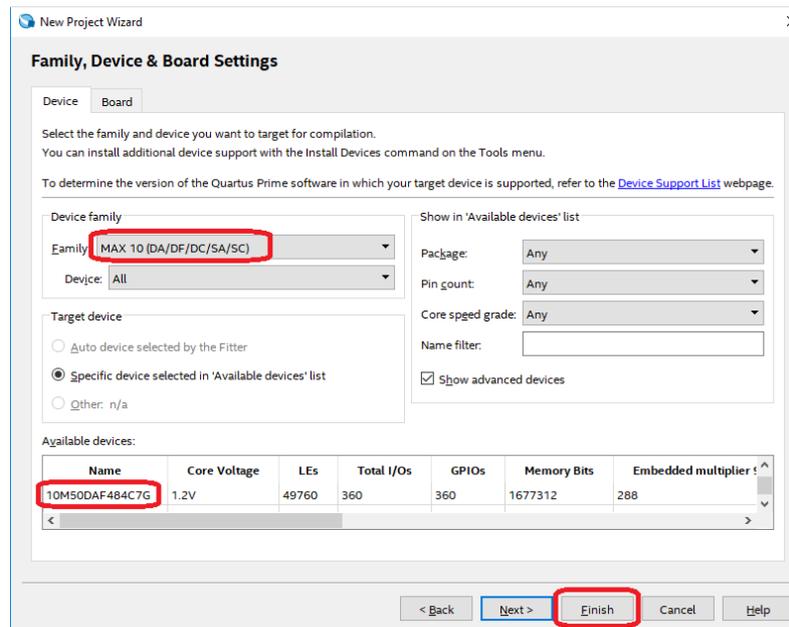
Figure 3-17 Connections between the 7-segment display HEX0 and the MAX 10 FPGA

Table 3-6 Pin Assignment of 7-segment Displays

Signal Name	FPGA Pin No.	Description	I/O Standard
HEX00	PIN_C14	Seven Segment Digit 0[0]	3.3-V LVTTTL
HEX01	PIN_E15	Seven Segment Digit 0[1]	3.3-V LVTTTL
HEX02	PIN_C15	Seven Segment Digit 0[2]	3.3-V LVTTTL
HEX03	PIN_C16	Seven Segment Digit 0[3]	3.3-V LVTTTL
HEX04	PIN_E16	Seven Segment Digit 0[4]	3.3-V LVTTTL
HEX05	PIN_D17	Seven Segment Digit 0[5]	3.3-V LVTTTL
HEX06	PIN_C17	Seven Segment Digit 0[6]	3.3-V LVTTTL
HEX07	PIN_D15	Seven Segment Digit 0[7], DP	3.3-V LVTTTL
HEX10	PIN_C18	Seven Segment Digit 1[0]	3.3-V LVTTTL
HEX11	PIN_D18	Seven Segment Digit 1[1]	3.3-V LVTTTL
HEX12	PIN_E18	Seven Segment Digit 1[2]	3.3-V LVTTTL
HEX13	PIN_B16	Seven Segment Digit 1[3]	3.3-V LVTTTL

HEX14	PIN_A17	Seven Segment Digit 1[4]	3.3-V LVTTTL
HEX15	PIN_A18	Seven Segment Digit 1[5]	3.3-V LVTTTL
HEX16	PIN_B17	Seven Segment Digit 1[6]	3.3-V LVTTTL
HEX17	PIN_A16	Seven Segment Digit 1[7] , DP	3.3-V LVTTTL
HEX20	PIN_B20	Seven Segment Digit 2[0]	3.3-V LVTTTL
HEX21	PIN_A20	Seven Segment Digit 2[1]	3.3-V LVTTTL
HEX22	PIN_B19	Seven Segment Digit 2[2]	3.3-V LVTTTL
HEX23	PIN_A21	Seven Segment Digit 2[3]	3.3-V LVTTTL
HEX24	PIN_B21	Seven Segment Digit 2[4]	3.3-V LVTTTL
HEX25	PIN_C22	Seven Segment Digit 2[5]	3.3-V LVTTTL
HEX26	PIN_B22	Seven Segment Digit 2[6]	3.3-V LVTTTL
HEX27	PIN_A19	Seven Segment Digit 2[7] , DP	3.3-V LVTTTL
HEX30	PIN_F21	Seven Segment Digit 3[0]	3.3-V LVTTTL
HEX31	PIN_E22	Seven Segment Digit 3[1]	3.3-V LVTTTL
HEX32	PIN_E21	Seven Segment Digit 3[2]	3.3-V LVTTTL
HEX33	PIN_C19	Seven Segment Digit 3[3]	3.3-V LVTTTL
HEX34	PIN_C20	Seven Segment Digit 3[4]	3.3-V LVTTTL
HEX35	PIN_D19	Seven Segment Digit 3[5]	3.3-V LVTTTL
HEX36	PIN_E17	Seven Segment Digit 3[6]	3.3-V LVTTTL
HEX37	PIN_D22	Seven Segment Digit 3[7] , DP	3.3-V LVTTTL
HEX40	PIN_F18	Seven Segment Digit 4[0]	3.3-V LVTTTL
HEX41	PIN_E20	Seven Segment Digit 4[1]	3.3-V LVTTTL
HEX42	PIN_E19	Seven Segment Digit 4[2]	3.3-V LVTTTL
HEX43	PIN_J18	Seven Segment Digit 4[3]	3.3-V LVTTTL
HEX44	PIN_H19	Seven Segment Digit 4[4]	3.3-V LVTTTL
HEX45	PIN_F19	Seven Segment Digit 4[5]	3.3-V LVTTTL
HEX46	PIN_F20	Seven Segment Digit 4[6]	3.3-V LVTTTL
HEX47	PIN_F17	Seven Segment Digit 4[7] , DP	3.3-V LVTTTL
HEX50	PIN_J20	Seven Segment Digit 5[0]	3.3-V LVTTTL
HEX51	PIN_K20	Seven Segment Digit 5[1]	3.3-V LVTTTL
HEX52	PIN_L18	Seven Segment Digit 5[2]	3.3-V LVTTTL
HEX53	PIN_N18	Seven Segment Digit 5[3]	3.3-V LVTTTL
HEX54	PIN_M20	Seven Segment Digit 5[4]	3.3-V LVTTTL
HEX55	PIN_N19	Seven Segment Digit 5[5]	3.3-V LVTTTL
HEX56	PIN_N20	Seven Segment Digit 5[6]	3.3-V LVTTTL
HEX57	PIN_L19	Seven Segment Digit 5[7] , DP	3.3-V LVTTTL

## בחר רכיב המתאים לכרטיס DE10-Lite

תוכנית למונה

```

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;

entity count0to9_1sec is
    port (clk, clear: in std_logic;
          LED: out STD_LOGIC_VECTOR (6 downto 0);
          q: buffer std_logic_vector (3 downto 0));
end;

architecture arc_count of count0to9_1sec is
    constant Fosc: integer := 50000000 ;
    signal en_1sec: std_logic := '0';
begin
    process (clk, clear)
        variable cnt: integer range 0 to (Fosc-1) ;
    begin
        if clear='0' then
            cnt:=0;
            en_1sec<='0';
        elsif rising_edge (clk) then
            if cnt < (Fosc-1) then
                cnt:=cnt+1;
                en_1sec<='0';
            else
                en_1sec<='1';
                cnt:=0;
            end if;
        end if;
    end process;

    process (clk, clear)
    begin

```

```

    if clear='0' then
        q<="0000";
    elsif rising_edge(clk) then
        if en_1sec='1' then
            if q<"1001" then q<=q+1;
            else q<="0000"; end if;
        end if;
    end if;
end process;

-- segment encoding
--      0
--      ---
--  5 |   | 1
--      --- <- 6
--  4 |   | 2
--      ---
--      3
with Q select
LED    <=
"1111001" when "0001",  --1
"0100100" when "0010",  --2
"0110000" when "0011",  --3
"0011001" when "0100",  --4
"0010010" when "0101",  --5
"0000010" when "0110",  --6
"1111000" when "0111",  --7
"0000000" when "1000",  --8
"0010000" when "1001",  --9
"0001000" when "1010",  --A
"0000011" when "1011",  --b
"1000110" when "1100",  --C
"0100001" when "1101",  --d
"0000110" when "1110",  --E
"0001110" when "1111",  --F
"1000000" when others;  --0
end;
```

### קביעת פינים

- KEY0 PIN\_B8 - clear לחצן
- MAX10\_CLK1\_50 PIN\_P11 - שעון clk
- מוצא Q – 4 לדים

LEDR0	PIN_A8	LED [0]
LEDR1	PIN_A9	LED [1]
LEDR2	PIN_A10	LED [2]
LEDR3	PIN_B10	LED [3]

- תצוגה 7 segment

Signal Name	FPGA Pin No.	Description
HEX00	PIN_C14	Seven Segment Digit 0[0]
HEX01	PIN_E15	Seven Segment Digit 0[1]
HEX02	PIN_C15	Seven Segment Digit 0[2]
HEX03	PIN_C16	Seven Segment Digit 0[3]
HEX04	PIN_E16	Seven Segment Digit 0[4]
HEX05	PIN_D17	Seven Segment Digit 0[5]
HEX06	PIN_C17	Seven Segment Digit 0[6]

דרך נוספת לתכנן בצורה היררכית, לפרק לחלקים, לתכנן כל חלק בנפרד ולחברם בצורה גרפית

לפתוח פרויקט חדש ולהגדיר את הרכיב

לפתוח קובץ VHDL חדש ולשמור בשם של ה-entity

```
bcd_7seg.vhd .1
count0to9.vhd .2
Generator_1Hz.vhd .3
```

מפענח BCD לתצוגה 7segment

```
library IEEE;
use IEEE.std_logic_1164.all;

entity bcd_7seg is
  port(
    LED: out STD_LOGIC_VECTOR (6 downto 0);
    Din: in std_logic_vector(3 downto 0));
end;

architecture arc_7seg of bcd_7seg is

begin

  -- segment encoding
  --      0
  --      ---
  --  5 |  | 1
  --      ---  <- 6
  --  4 |  | 2
  --      ---
  --      3

  with Din select
  LED    <=
    "1111001" when "0001",  --1
    "0100100" when "0010",  --2
    "0110000" when "0011",  --3
    "0011001" when "0100",  --4
    "0010010" when "0101",  --5
    "0000010" when "0110",  --6
    "1111000" when "0111",  --7
    "0000000" when "1000",  --8
    "0010000" when "1001",  --9
    "0001000" when "1010",  --A
    "0000011" when "1011",  --b
    "1000110" when "1100",  --C
    "0100001" when "1101",  --d
    "0000110" when "1110",  --E
    "0001110" when "1111",  --F
    "1000000" when others;  --0

end;
```

```

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;

entity Count0to9 is
    port (clk ,clear , en_1s: in std_logic;
          q: buffer std_logic_vector(3 downto 0));
end;

architecture arc_count of Count0to9 is
begin
    process (clk,clear)
    begin
        if clear='0' then
            q<="0000";
        elsif rising_edge(clk) then
            if en_1s = '1' then
                if q<"1001" then q<=q+1;
                else q<="0000";
                end if;
            end if;
        end if;
    end process;
end;

```

מאפשר מניה כל שניה

```

library IEEE;
use IEEE.std_logic_1164.all;

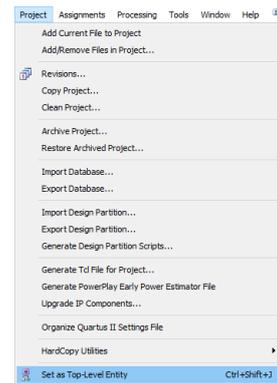
entity Generator_1Hz is
    port (clk ,clear: in std_logic;

          en_1sec: buffer std_logic);
end;

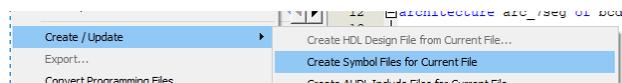
architecture arc_1Hz of Generator_1Hz is
    constant Fosc: integer := 50000000 ;
begin
    process (clk)
        variable cnt: integer range 0 to (Fosc-1) ;
    begin
        if clear='0' then
            cnt:=0;
            en_1sec<='0';
        elsif rising_edge(clk) then
            if cnt < (Fosc-1) then
                cnt:=cnt+1;
                en_1sec<='0';
            else
                en_1sec<='1';
                cnt:=0;
            end if;
        end if;
    end process;
end;

```

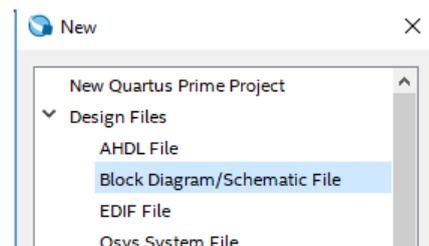
כל אחד מהם לכתוב את הקוד , להפוך ל- top level ולבצע קומפילציה



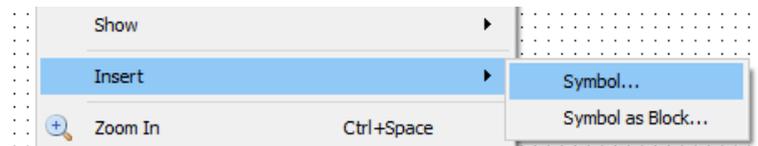
לבנות מודול גרפי לכל חלק



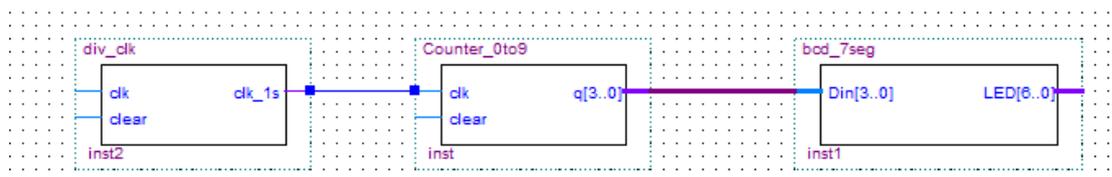
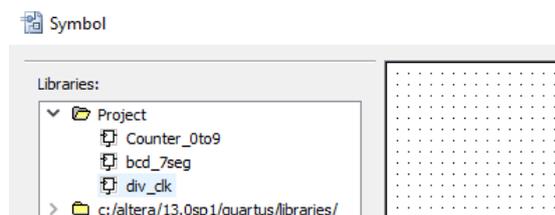
לפתוח קובץ בלוקים חדש ולשמור בשם top\_counter



לחצן ימני על מסך הגרפי

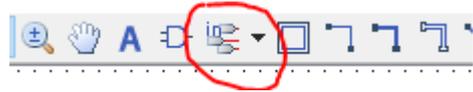


נקבל רשימת המודולים שבנינו, להוסיף אותם לשרטוט

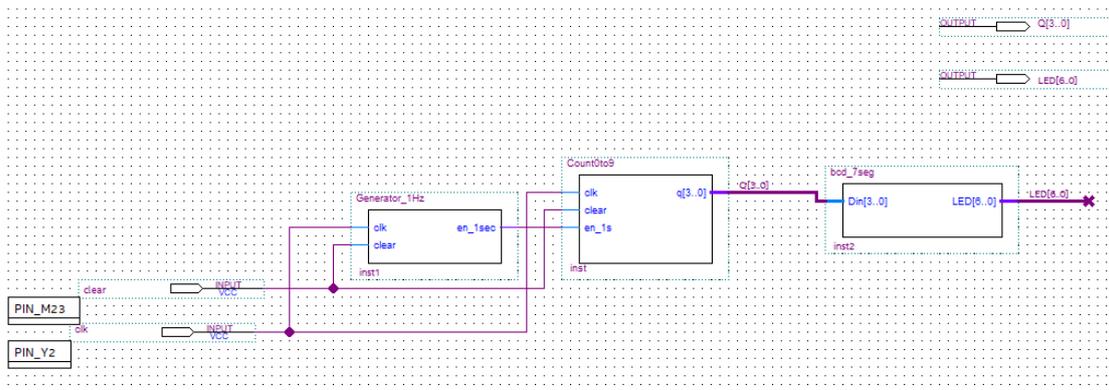


לחבר בין המודולים

להוסיף input ו- output



שם על ה-BUS – לסמן את BUS ולחצן ימני <- properties ולרשום שם בצורה LED[6..0]



להפוך את הקובץ ל- TOP LEVEL ולבצע קומפילציה

Project -> set as top level

- לקבוע פינים
- קומפילציה וצריבה