CPLD 設計作業

範圍：多工掃瞄

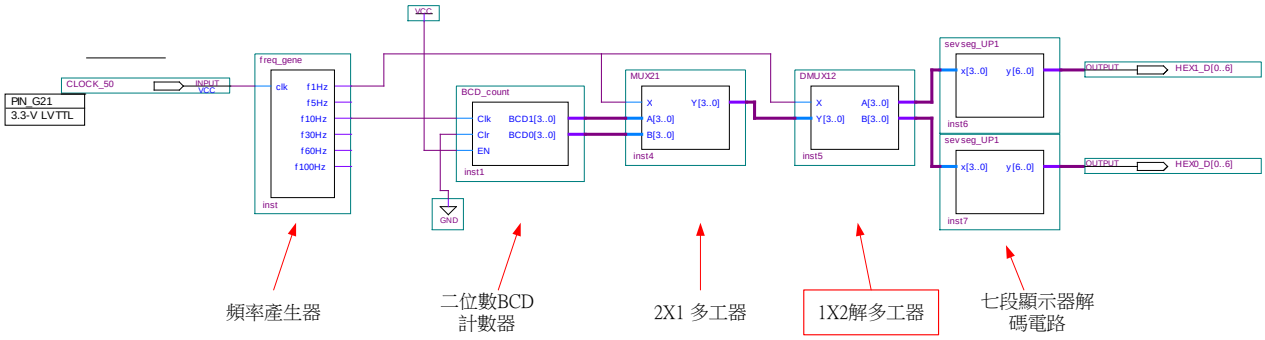
銘傳大學電腦與通訊工程系

班　　級	電通四甲
姓　　名	余采潔
學　　號	05051115
作業成果	應繳作業共 2 題，每題 50 分 我共完成 2 題，應得 100 分
授課教師	陳慶逸

■ 請確實填寫自己寫完成題數，填寫不實者(如上傳與作業明顯無關的答案，或是計算題數有誤者)，本次作業先扣 50 分。

第一題

實現二位數多工掃瞄。

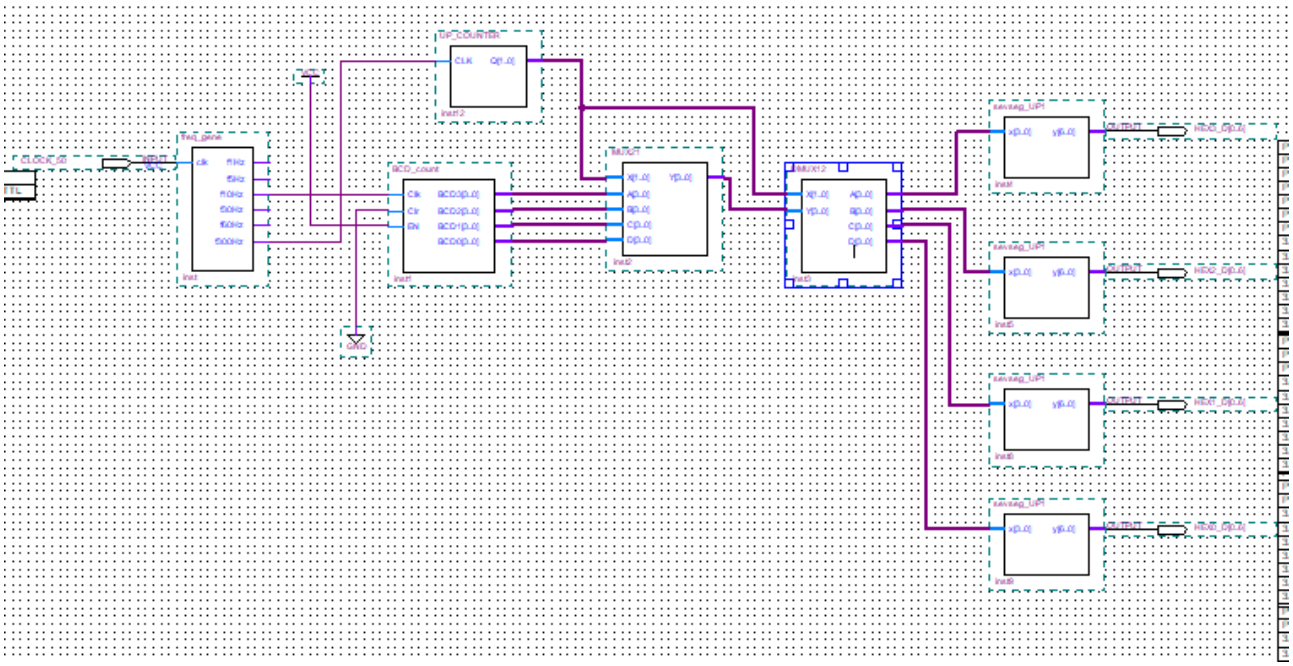


https://youtu.be/6-cwicaD7q4

第二題

利用 DE0 實現一個多工掃瞄式的 4 位數 BCD 計數器

https://youtu.be/8GZ4NMVUEzM



slowCLK

```
library ieee;
use ieee.std_logic_1164.all;

entity slowCLK is
  generic(divisor:integer:=50000000);
port(    clockIN : in    std_logic;
                    clockOUT: out std_logic);
end slowCLK;

architecture arch of slowCLK is
signal PULSE : std_logic;
```

```
begin
---------- clk divider ---------------
process(clockIN)
variable counter,divisor2 : integer range 0 to divisor;
begin
   divisor2:=divisor/2;
----------- up counter -----------------
       if (clockIN 'event and clockIN ='1') then
          if counter = divisor then
             counter := 1;
          else
             counter := counter + 1;
          end if;
       end if;
   ----- clk generator -----
       if (clockIN 'event and clockIN ='1') then
          if (( counter= divisor2) or (counter = divisor))then
             PULSE <= not PULSE ;
          end if;
       end if;
       clockOUT <= PULSE ;
   end process;
end arch;
```

```
--實驗名稱：frequency generator

--檔案名稱：freq_gene.vhd

--功　　能：產生 1Hz, 5Hz, 10Hz,30Hz, 60Hz,100Hz

--日　　期：2013.12.2

----------------------------------------------------------------------
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;

entity freq_gene is
port(
        clk : in std_logic;
        f1Hz,f5Hz,f10Hz,f30Hz,f60Hz,f300Hz: out std_logic
      );
end freq_gene;

architecture arch of freq_gene is

component slowCLK
   generic(divisor:integer:=8);
port( clockIN : in    std_logic;
          clockOUT: out std_logic);
end component;
```

```
begin

u1: slowCLK
      generic map(50000000)     --1Hz
      port map(clk,f1Hz);
u2: slowCLK
      generic map(10000000)     --5Hz
      port map(clk,f5Hz);
u3: slowCLK
      generic map(5000000)     --10Hz
      port map(clk,f10Hz);
u4: slowCLK
      generic map(1666666)     --30Hz
      port map(clk,f30Hz);
u5: slowCLK
      generic map(833333)      --60Hz
      port map(clk,f60Hz);
u6: slowCLK
      generic map(125000)      --100Hz
      port map(clk,f300Hz);
end arch;
```

UP_COUNTER

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
```

```
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY UP_COUNTER is
PORT( CLK: IN STD_LOGIC;
Q    : OUT STD_LOGIC_VECTOR(1 DOWNTO 0));
END UP_COUNTER;

ARCHITECTURE a OF UP_COUNTER IS
SIGNAL QN : STD_LOGIC_VECTOR(1 DOWNTO 0);
BEGIN
     PROCESS (CLK)
     BEGIN
          IF CLK'event AND CLK='1' THEN
                    QN <= QN+1;
          END IF;
     END PROCESS;
     Q<=QN;
END a;
```

MUX21

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

ENTITY MUX21 IS
PORT ( X    : IN    STD_LOGIC_vector(1 downto 0);
          A,B,C,D: IN STD_LOGIC_vector(3 downto 0);
```

```
        Y: OUT STD_LOGIC_vector(3 downto 0));
END MUX21;

ARCHITECTURE a OF MUX21 IS
BEGIN

      Y <= A when X="00" else
           B when X="01" else
           C when X="10" else
           D ;
END a;
```

BCD_count

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;
USE ieee.std_logic_unsigned.all ;

ENTITY BCD_count IS
PORT ( Clk : IN STD_LOGIC ;
           Clr, EN : IN STD_LOGIC ;
           BCD3, BCD2, BCD1, BCD0 : BUFFER STD_LOGIC_VECTOR(3
DOWNTO 0) ) ;
  END BCD_count ;

ARCHITECTURE Behavior OF BCD_count IS
```

```
BEGIN
PROCESS (Clk)
BEGIN
   IF Clk'EVENT AND Clk = '1' THEN
        IF Clr = '1' THEN
             BCD3 <= "0000" ; BCD2 <= "0000" ;BCD1 <= "0000" ; BCD0 <=
"0000" ;
        ELSIF EN = '1' THEN
             IF BCD0 = "1001" THEN
                BCD0 <= "0000" ;
                IF BCD1 = "1001" THEN
                     BCD1 <= "0000";
                     IF BCD2 = "1001" THEN
                     BCD2 <= "0000" ;
                     IF BCD3 = "1001" THEN
                          BCD3 <= "0000";
                     ELSE
                          BCD3 <= BCD3 + '1' ;
                     END IF ;
                  ELSE
                     BCD2 <= BCD2 + '1' ;
                  END IF ;
                ELSE
                     BCD1 <= BCD1 + '1' ;
                END IF ;
             ELSE
                BCD0 <= BCD0 + '1' ;
```

```
            END IF ;
        END IF ;
    END IF;
END PROCESS;
END Behavior ;
```

DMUX12

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

ENTITY DMUX12 IS
PORT (X    : IN    STD_LOGIC_vector(1 downto 0);
        Y: IN STD_LOGIC_vector(3 downto 0);
        A,B,C,D: OUT STD_LOGIC_vector(3 downto 0));
END DMUX12;

ARCHITECTURE a OF DMUX12 IS
BEGIN

    A <= Y when X="00" else "1111";
    B <= Y when X="01" else "1111";
    C <= Y when X="10" else "1111";
    D <= Y when X="11" else "1111";

END a;
```