CPLD 設計作業

銘傳大學電腦與通訊工程系

班　　級	電通二乙
組　　別	第 27 組
組員姓名/學號	陳韋澔/05050032
組員姓名/學號	陳奕璁/05050536
作業撰寫人	陳韋澔/05050032
授課教師	陳慶逸
實習成果	本次應繳作業共 2 題，完成 2 題
時　　間	2017/12/29

第一題

◘ 問題描述: 驗證 PWM 程式，利用 SW[7]-SW[0]控制 LEDG[0]的輸出亮度。

PWM 程式

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

entity pwm is
port(
        CLOCK_50 : in    std_logic;                          --系統頻率
        SW     : in    std_logic_vector(7 downto 0); --PWM 控制訊號
        pwm: out std_logic                                  --PWM 訊號輸出
      );
end pwm;

architecture a of pwm is
   signal B:std_logic_vector(7 downto 0);
begin

   ----- 下數計數器 -----
   process(CLOCK_50)
   begin
      if CLOCK_50'event and CLOCK_50='1' then
```

```
        B <= B-1;
      end if;
   end process;

   --比較器
   pwm<='1' when SW > B else '0';

end a;
```

成果: https://www.youtube.com/watch?v=S7YRiCqYM-M

第二題

題目:設計一個自動切換上下數的計數器 (00000000->00000001->…->11111111->11111110->….00000000->…)，用以控制 pwm 輸出的明亮變化。

頻率器

```
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
```

大頻率器

```
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,f100Hz: 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(500000)       --100Hz
        port map(clk,f100Hz);
end arch;
```

上下數計數器

```
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(7 DOWNTO 0));
END UP_COUNTER;
ARCHITECTURE a OF UP_COUNTER IS
SIGNAL QN : STD_LOGIC_VECTOR(7 DOWNTO 0);
SIGNAL SN : STD_LOGIC;
BEGIN
      PROCESS (CLK)
      BEGIN
          IF CLK'event AND CLK='1' THEN
          IF SN='0' THEN
```

```
        IF QN="11111111" THEN
         SN<='1';
        ELSE
         QN<=QN+1;
        END IF;
        ELSE
        IF QN="00000000" THEN
         SN<='0';
        ELSE
         QN<=QN-1;
        END IF;
        END IF;
        END IF;
    END PROCESS;
    Q<=QN;
END a;
```

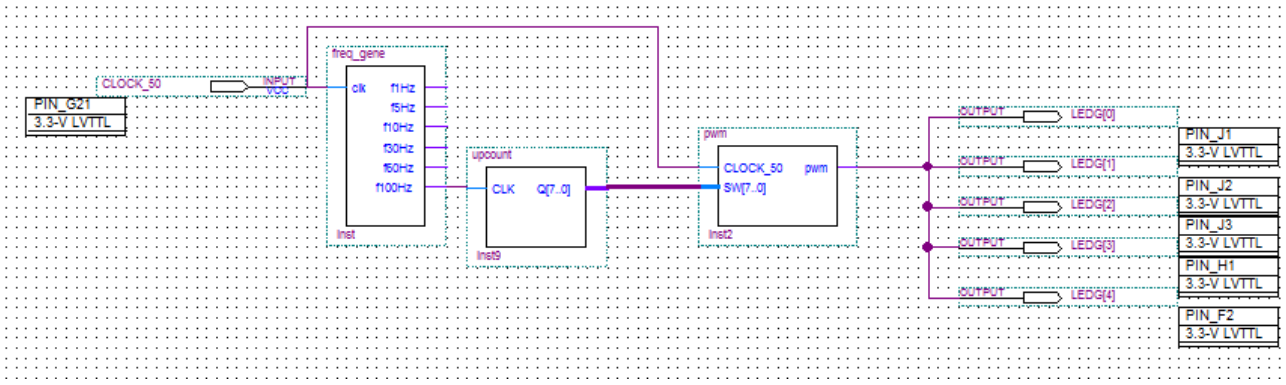
PWM 程式

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

entity pwm is
port(
        CLOCK_50 : in    std_logic;                    --系統頻率
```

```
        SW     : in    std_logic_vector(7 downto 0); --PWM 控制訊號
        pwm: out std_logic                                --PWM 訊號輸出
    );
end pwm;

architecture a of pwm is
  signal B:std_logic_vector(7 downto 0);
begin

  ----- 下數計數器 -----
  process(CLOCK_50)
  begin
    if CLOCK_50'event and CLOCK_50='1' then
      B <= B-1;
    end if;
  end process;

  --比較器
  pwm<='1' when SW > B else '0';
end a;
```

成果:

https://www.youtube.com/watch?v=eUaousXJSPY


CLOCK_50
INPUT
VCC
PIN_G21
3.3-V LVTTL
freq_gene
clk
f1Hz
f5Hz
f10Hz
f30Hz
f50Hz
f100Hz
inst
upcount
CLK
Q[7..0]
inst9
pwm
CLOCK_50
pwm
SW[7..0]
inst2
OUTPUT
LEDG[0]
LEDG[1]
LEDG[2]
LEDG[3]
LEDG[4]
PIN_J1
PIN_J2
PIN_J3
PIN_H1
PIN_F2
3.3-V LVTTL

影片連結: