

Dht11_Driver.vhd

```
--Dht11_Driver
--直接輸出濕度 (DHT11_DBoH) 及溫度 (DHT11_DBoT):integer(0~255:8bit)
--107.01.01 版
--EP3C16Q240C8 50MHz

Library IEEE;                                --連結零件庫
Use IEEE.std_logic_1164.all;                  --引用套件
Use IEEE.std_logic_unsigned.all;              --引用套件

entity Dht11_Driver is
    port (DHT11_CLK,DHT11_RESET:in std_logic;
          --DHT11_CLK:781250Hz (50MHz/2^6:1.28us:FD(5)) 操作速率,重置
          DHT11_D_io:inout std_logic;          --DHT11 i/o
          DHT11_DBo:out std_logic_vector(7 downto 0);
          --DHT11_driver 資料輸出
          DHT11_RDp:in integer range 0 to 7;--資料讀取指標
          DHT11_tryN:in integer range 0 to 7;--錯誤後嘗試幾次
          DHT11_ok,DHT11_S:buffer std_logic;
          --DHT11_driver 完成作業旗標, 錯誤信息
          DHT11_DBoH,DHT11_DBoT:out integer range 0 to 255);
          --直接輸出濕度及溫度
end entity Dht11_Driver;

architecture Albert of Dht11_Driver is
    signal S_B,bit01,response:std_logic;      --start bit,接收位元
    signal ss:std_logic_vector(1 downto 0);--執行狀態
    signal isdata:integer range 0 to 3;        --資料狀態
    signal dp,d8:integer range 0 to 7;         --資料位元操作指標
    signal dbit:std_logic_vector(6 downto 0);--byte
    signal chK_SUM:std_logic_vector(7 downto 0);--查和
    type DDataT is array(0 to 4) of std_logic_vector(7 downto 0);
    --資料緩衝區格式
    signal dd:DDataT;    --資料緩衝區
    signal tryNN:integer range 0 to 7;         --錯誤後嘗試幾次
    signal Timeout:std_logic_vector(21 downto 0);--timeout 計時器
    signal tryDelay:integer range 0 to 31;
begin

    DHT11_DBoH<=conv_integer(dd(4));--直接輸出濕度(integer)
    DHT11_DBoT<=conv_integer(dd(2));--直接輸出溫度(integer)

    --DHT11_DBo 由 DHT11_RDp 選取輸出項
    DHT11_DBo<=dd(DHT11_RDp) when DHT11_RDp<5 else
        chK_SUM when DHT11_RDp=5 else (others=>'1');--上傳資料

    DHT11_D_io<='Z' when DHT11_RESET='0' or S_B='1' else '0';
    --DHT11 data io 操作

    DHT11:process(DHT11_CLK,DHT11_RESET)
begin
```

```

if DHT11_RESET= '0' then
    S_B<='0'; --start bit
    dp<=4; --讀取 5byte
    d8<=7; --讀取 8bit
    isdata<=2; --資料狀態
    DHT11_ok<='0'; --未完成作業
    DHT11_S<='0'; --解除作業失敗
    tryNN<=DHT11_tryN; --錯誤後嘗試幾次
    ss<="00"; --執行狀態由 1 開始
    Timeout<=(others=>'0'); --timeout 計時器歸零
    tryDelay<=11;
    --11:約 2.5ms,12:約 5ms,13:約 10ms,14:約 21ms~18ms,15:約 42ms~18ms
elseif rising_edge(DHT11_CLK) and DHT11_ok='0' then
    Timeout<=Timeout+1; --計時
    case ss is
        --restart or Send request
        when "00"=> --產生封包起始信號(主機要求讀取溫濕度資料)
            --重啟 (restart:D_io->'Z')or(start bit:D_io->'0')
            if Timeout(tryDelay)='1' then
                --start bit (最好能在 2ms 以上較穩定) Request DHT11
                tryDelay<=11;
                --11:約 2.5ms,12:約 5ms,13:約 10ms,14:約 21ms,15:約 42ms
                S_B<=not S_B; --輸出控制位元
                Timeout<=(others=>'0'); --重起計時
                ss<="0" & not S_B; --執行狀態下一步
                chK_SUM<=(others=>'0'); --查和歸零
                response<='0'; --愈時內定選擇
            end if;

            --wait DHT11 Response pull low
            when "01"=>--等待 DHT11 回應(約 80μs 低態)
                if DHT11_D_io='0' then
                    Timeout<=(others=>'0');
                    if isdata=0 then --reciver bit
                        d8<=d8-1; --應接收位元數遞減
                        if d8=0 then --已收到 8bit
                            dp<=dp-1; --應接收筆數遞減
                            dd(dp)<=dbit & bit01;
                            --接收位元及存入資料緩衝區
                            if dp<4 then
                                chK_SUM<=chK_SUM+dd(dp+1); --計算查和
                            end if;
                            ss<="10"; --執行狀態下一步 pull high
                        else
                            dbit<=dbit(5 downto 0)&bit01;--接收位元
                            ss<="10"; --執行狀態下一步 pull high
                        end if;
                    else
                        isdata<=isdata-1;
                        ss<="10"; --執行狀態下一步 pull high
                    end if;
                elseif Timeout=38 then --約 49us
                    bit01<='1'; --接收位元 0-->1

```

```

--約 Response(error)21ms>11~13ms
--或 DHT11 No data Response(error) 約 164us
elseif (Timeout(14)='1'and response='0')
    or(Timeout(7)='1'and response='1') then
    ss<="11";          --執行狀態下一步(錯誤處理)
end if;

--wait DHT11 Response pull high
when "10"=>
    if DHT11_D_io='1' then
        Timeout<=(others=>'0');--重起計時
        bit01<='0';          --接收位元預設 0
        if dp=7 then --(已讀取 40bit)stop bit
            if chK_SUM=dd(0) then
                DHT11_ok<='1'; --作業已正確完成
            else
                ss<="11";          --執行狀態下一步(錯誤處理)
            end if;
        else
            ss<="01";          --執行狀態下一步
        end if;
    elseif Timeout(7)='1' then--DHT11 No Response(error) 7
        ss<="11";          --執行狀態下一步(錯誤處理) --約 164us
    end if;

    when others=>          --"11"錯誤處理
        if tryNN/=0 then
            tryNN<=tryNN-1;          --嘗試錯誤數遞減
            Timeout<=(others=>'0');--重起計時
            dp<=4;          --應接收筆數
            d8<=7;          --應接收位元數
            isdata<=2;      --開始階段
            tryDelay<=20;    --約暫停 1.4s
            ss<="00";        --restart
        else
            DHT11_ok<='1';          --作業已完成
            DHT11_s<='1';          --作業失敗
        end if;
    end case;
end if;
end process;

end Albert;

```

CH10_DHT11_1.vhd

```

--DHT11 溫濕度感測器測試:1 wire+中文 LCM 顯示
--107.01.01 版
--EP3C16Q240C8 50MHz LEs:15,408 PINs:161 ,gckp31 ,rstP99

```

```

Library IEEE;                                --連結零件庫
Use IEEE.std_logic_1164.all;                 --引用套件
Use IEEE.std_logic_unsigned.all;             --引用套件
use ieee.std_logic_arith.all;                 --引用套件


entity CH10_DHT11_1 is
port (gckp31,rstP99:in std_logic;             --系統頻率,系統 reset
      --DHT11
      DHT11_D_io:inout std_logic;             --DHT11 i/o

      --LCD 4bit 介面
      DB_io:inout std_logic_vector(3 downto 0);
      RSo,RWo,Eo:out std_logic
    );
end entity CH10_DHT11_1;


architecture Albert of CH10_DHT11_1 is
  component DHT11_driver is
    port (DHT11_CLK,DHT11_RESET:in std_logic;
          --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
          DHT11_D_io:inout std_logic;           --DHT11 i/o
          DHT11_DBo:out std_logic_vector(7 downto 0);
          --DHT11_driver 資料輸出
          DHT11_RDp:in integer range 0 to 7;     --資料讀取指標
          DHT11_tryN:in integer range 0 to 7; --錯誤後嘗試幾次
          DHT11_ok,DHT11_S:buffer std_logic;
          --DHT11_driver 完成作業旗標,錯誤信息
          DHT11_DBoH,DHT11_DBoT:out integer range 0 to 255);
          --直接輸出濕度及溫度
    end component DHT11_driver;

  signal DHT11_CLK,DHT11_RESET:std_logic;
  --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
  signal DHT11_DBo:std_logic_vector(7 downto 0);
  --DHT11_driver 資料輸出
  signal DHT11_RDp:integer range 0 to 7;         --資料讀取指標 5~0
  signal DHT11_tryN:integer range 0 to 7:=3; --錯誤後嘗試幾次
  signal DHT11_ok,DHT11_S:std_logic;
  --DHT11_driver 完成作業旗標,錯誤信息
  signal DHT11_DBoH,DHT11_DBoT:integer range 0 to 255;
  --直接輸出濕度及溫度


  --中文 LCM 4bit driver(WG14432B5)
  component LCM_4bit_driver is
    port (LCM_CLK,LCM_RESET:in std_logic; --操作速率,重置
          RS,RW:in std_logic;             --暫存器選擇,讀寫旗標輸入
          DBi:in std_logic_vector(7 downto 0); --LCM_4bit_driver 資料輸入
          DBo:out std_logic_vector(7 downto 0); --LCM_4bit_driver 資料輸出
          DB_io:inout std_logic_vector(3 downto 0); --LCM DATA BUS 介面
          RSo,RWo,Eo:out std_logic;         --LCM 暫存器選擇,讀寫,致能介面
          LCMok,LCM_S:out boolean --LCM_4bit_driver 完成,錯誤旗標
    );

```

```

end component;

signal LCM_RESET,RS,RW:std_logic;
--LCM_4bit_driver 重置,LCM 暫存器選擇,讀寫旗標
signal DBi,DBo:std_logic_vector(7 downto 0);
--LCM_4bit_driver 命令或資料輸入及輸出
signal LCMok,LCM_S:boolean; --LCM_4bit_driver 完成作業旗標,錯誤信息

signal FD:std_logic_vector(24 downto 0);--除頻器
signal times:integer range 0 to 2047; --計時器

--中文 LCM 指令&資料表格式:
--(總長,指令數,指令...資料.....)
--英數型 LCM 4 位元界面,2 列顯示

type LCM_T is array (0 to 20) of std_logic_vector(7 downto 0);
constant LCM_IT:LCM_T:=(X"0F",X"06",--中文型 LCM 4 位元界面
"00101000","00101000","00101000",--4 位元界面
"00000110","00001100","00000001",
--ACC+1 顯示幕無移位,顯示幕 on 無游標無閃爍,清除顯示幕
X"01",X"48",X"65",X"6C",X"6C",X"6F",X"21",X"20",X"20",
X"20",X"20",X"20",X"20");--Hello!

--LCM=1:第一列顯示區 DHT11 測濕度 %RH
signal LCM_1:LCM_T:=(X"15",X"01", --總長,指令數
"00000001", --清除顯示幕
--第 1 列顯示資料
X"44",X"48",X"54",X"31",X"31",X"20",X"B4",X"FA",X"C0",
X"E3",X"AB",X"D7",X"3D",X"30",X"30",X"25",X"52",X"48");
--DHT11 測濕度 %RH

--LCM=1:第二列顯示區 DHT11 測溫度 °C
signal LCM_12:LCM_T:=(X"15",X"01", --總長,指令數
"10010000", --設第二列 ACC 位置
--第 2 列顯示資料
X"44",X"48",X"54",X"31",X"31",X"20",X"B4",X"FA",X"B7",
X"C5",X"AB",X"D7",X"3D",X"30",X"30",X"20",X"A2",X"4A");
--DHT11 測溫度 °C

--LCM=2:第一列顯示區 DHT11 資料讀取失敗
signal LCM_2:LCM_T:=(X"15",X"01", --總長,指令數
"00000001", --清除顯示幕
--第 1 列顯示資料
X"44",X"48",X"54",X"31",X"31",X"20",X"B8",X"EA",X"AE",
X"C6",X"C5",X"AA",X"A8",X"FA",X"A5",X"A2",X"B1",X"D1");
--DHT11 資料讀取失敗

signal LCM_com_data,LCM_com_data2:LCM_T;
signal LCM_INI:integer range 0 to 31;
signal LCMP_RESET,LN,LCMPok:std_logic;
signal LCM,LCMx:integer range 0 to 7;

begin

```

```

DHT11_CLK<=FD(5);--DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率
U2: DHT11_driver port map(
    DHT11_CLK,DHT11_RESET,
    --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
    DHT11_D_io,          --DHT11 i/o
    DHT11_DBo,          --DHT11_driver 資料輸出
    DHT11_RDp,          --資料讀取指標
    DHT11_tryN,          --錯誤後嘗試幾次
    DHT11_ok,DHT11_S,DHT11_DBoH,DHT11_DBoT);
    --DHT11_driver 完成作業旗標,錯誤信息,直接輸出濕度及溫度
--中文 LCM
LCMset: LCM_4bit_driver port map(
    D(7),LCM_RESET,RS,RW,DBi,DBo,DB_io,RSo,RWo,Eo,LCMok,LCM_S);
    --LCM 模組

DHT11P_Main:process(FD(17))
begin
    if rstP99='0' then          --系統重置
        DHT11_RESET<='0';      --DHT11 準備重新讀取資料
        LCM<=0;                --中文 LCM 初始化
        LCMP_RESET<='0';       --LCMP 重置
    elsif rising_edge(FD(17)) then
        LCMP_RESET<='1';       --LCMP 啟動顯示
        if LCMPOk='1' then
            if DHT11_RESET='0' then          --DHT11_driver 尚未啟動
                DHT11_RESET<='1';           --DHT11 資料讀取
                times<=400;                  --設定計時
            elsif DHT11_ok='1' then          --DHT11 讀取結束
                times<=times-1;              --計時
            if times=0 then                  --時間到
                LCM<=1;                     --中文 LCM 顯示測量值
                LCMP_RESET<='0';             --LCMP 重置
                DHT11_RESET<='0';           --DHT11 準備重新讀取資料
            elsif DHT11_S='1' then          --資料讀取失敗
                LCM<=2;                     --中文 LCM 顯示 DHT11 資料讀取失敗
            end if;
        end if;
    end if;
end if;
end process DHT11P_Main;

--DHT11 LCM 顯示
LCM_1(17)<="0011"&conv_std_logic_vector(DHT11_DBoH mod 10,4);
-- 擷取個位數
LCM_1(16)<="0011"&conv_std_logic_vector((DHT11_DBoH/10)mod 10,4);
-- 擷取十位數
LCM_12(17)<="0011"&conv_std_logic_vector(DHT11_DBoT mod 10,4);
-- 擷取個位數
LCM_12(16)<="0011"&conv_std_logic_vector((DHT11_DBoT/10)mod 10,4);
-- 擷取十位數

--中文 LCM 顯示器-----

```

```

--中文 LCM 顯示器
--指令&資料表格式:
--(總長,指令數,指令...資料.....
LCM_P:process(FD(0))
    variable SW:Boolean;          --命令或資料備妥旗標
begin
    if LCM/=LCMx or LCMP_RESET='0' then
        LCMx<=LCM;                --記錄選項
        LCM_RESET<='0';           --LCM 重置
        LCM_INI<=2;                --命令或資料索引設為起點
        LN<='0';                  --設定輸出 1 列
        case LCM is
            when 0=>
                LCM_com_data<=LCM_IT;  --LCM 初始化輸出第一列資料 Hello!
            when 1=>
                LCM_com_data<=LCM_1;    --輸出第一列資料
                LCM_com_data2<=LCM_12;  --輸出第二列資料
                LN<='1';                --設定輸出 2 列
            when others =>
                LCM_com_data<=LCM_2;    --輸出第一列資料
        end case;
        LCMPok<='0';               --取消完成信號
        SW:=False;                 --命令或資料備妥旗標
    elsif rising_edge(FD(0)) then
        if SW then                 --命令或資料備妥後
            LCM_RESET<='1';         --啟動 LCM_4bit_driver_delay
            SW:=False;              --重置旗標
        elsif LCM_RESET='1' then   --LCM_4bit_driver_delay 啟動中
            if LCMPok then          --等待 LCM_4bit_driver_delay 完成傳送
                LCM_RESET<='0';     --完成後 LCM 重置
            end if;
        elsif LCM_INI<LCM_com_data(0) and LCM_INI<LCM_com_data'length
then
            --命令或資料尚未傳完
            if LCM_INI<=(LCM_com_data(1)+1) then--選命令或資料暫存器
                RS<='0';            --Instruction reg
            else
                RS<='1';            --Data reg
            end if;
            RW<='0';                --LCM 寫入操作
            DBi<=LCM_com_data(LCM_INI);--載入命令或資料
            LCM_INI<=LCM_INI+1;     --命令或資料索引指到下一筆
            SW:=True;               --命令或資料已備妥
        else
            if LN='1' then          --設定輸出 2 列
                LN<='0';           --設定輸出 2 列取消
                LCM_INI<=2;         --命令或資料索引設為起點
                LCM_com_data<=LCM_com_data2;--LCM 輸出第二列資料
            else
                LCMPok<='1';        --執行完成
            end if;
        end if;
    end if;
end if;

```

```

end process LCM_P;

--除頻器--
Freq_Div:process(gckP31)          --系統頻率 gckP31:50MHz
begin
    if rstP99='0' then            --系統重置
        FD<=(others=>'0');        --除頻器:歸零
    elsif rising_edge(gckP31) then --50MHz
        FD<=FD+1;                --除頻器:2 進制上數(+1)計數器
    end if;
end process Freq_Div;

end Albert;

```

CH10_DHT11_2.vhd

```

--DHT11 溫濕度感測器測試:1 wire
--107.01.01 版
--EP3C16Q240C8 50MHz LEs:15,408 PINs:161 ,gckp31 ,rstP99

Library IEEE;                    --連結零件庫
Use IEEE.std_logic_1164.all;     --引用套件
Use IEEE.std_logic_unsigned.all; --引用套件
Use IEEE.numeric_std.all;        --引用套件

entity CH10_DHT11_2 is
port(gckp31,rstP99:in std_logic;  --系統頻率,系統 reset
    SW8_1,SW8_2:in std_logic_vector(7 downto 0);
    --指撥開關輸入:溫度設定,濕度設定
    --DHT11
    DHT11_D_io:inout std_logic;   --DHT11 i/o

    --DHT11 七段顯示器顯示輸出
    DHT11_scan:buffer unsigned(3 downto 0); --掃描信號
    D7data:out std_logic_vector(7 downto 0); --顯示資料
    D7xx_xx:out std_logic; --:

    --蜂鳴器輸出
    sound1,sound2:buffer std_logic
);
end entity CH10_DHT11_2;

architecture Albert of CH10_DHT11_2 is
    component DHT11_driver is
        port(DHT11_CLK,DHT11_RESET:in std_logic;
            --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
            DHT11_D_io:inout std_logic;    --DHT11 i/o
            DHT11_DBo:out std_logic_vector(7 downto 0);
            --DHT11_driver 資料輸出

```



```

        DHT11_RDp:in integer range 0 to 7;--資料讀取指標
        DHT11_tryN:in integer range 0 to 7;--錯誤後嘗試幾次
        DHT11_ok,DHT11_S:buffer std_logic;
        --DHT11_driver 完成作業旗標,錯誤信息
        DHT11_DBoH,DHT11_DBoT:out integer range 0 to 255);
        --直接輸出濕度及溫度
    end component DHT11_driver;
    signal DHT11_CLK,DHT11_RESET:std_logic;
    --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
    signal DHT11_DBo:std_logic_vector(7 downto 0);
    --DHT11_driver 資料輸出
    signal DHT11_RDp:integer range 0 to 7;        --資料讀取指標 5~0
    signal DHT11_tryN:integer range 0 to 7:=3; --錯誤後嘗試幾次
    signal DHT11_ok,DHT11_S:std_logic;
    --DHT11_driver 完成作業旗標,錯誤信息
    signal DHT11_DBoH,DHT11_DBoT:integer range 0 to 255;
    --直接輸出濕度及溫度

    signal FD:std_logic_vector(24 downto 0);    --除頻器
    signal scanP:integer range 0 to 3;          --位數取值指標
    signal HL,TL:std_logic;                    --濕度、溫度狀態
    signal D7sp:std_logic;                     --小數點
    signal Disp7S:std_logic_vector(6 downto 0);--顯示解碼

    type D7_data_T is array (0 to 3) of integer range 0 to 15;
    --DHT11 顯示值格式
    signal D7_data:D7_data_T:=(0,0,0,0);    --DHT11 顯示值
    signal times:integer range 0 to 2047;    --計時器

begin
DHT11_CLK<=FD(5);
--DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率
U2: DHT11_driver port map(
    DHT11_CLK,DHT11_RESET,
    --DHT11_CLK:781250Hz(50MHz/2^6:1.28us:FD(5))操作速率,重置
    DHT11_D_io,        --DHT11 i/o
    DHT11_DBo,        --DHT11_driver 資料輸出
    DHT11_RDp,        --資料讀取指標
    DHT11_tryN,        --錯誤後嘗試幾次
    DHT11_ok,DHT11_S,DHT11_DBoH,DHT11_DBoT);
    --DHT11_driver 完成作業旗標,錯誤信息,直接輸出濕度及溫度

DHT11P_Main:process(FD(17))
begin
    if rstP99='0' then        --系統重置
        DHT11_RESET<='0';    --DHT11 準備重新讀取資料
        D7xx_xx<='1';        --:不亮
    elsif rising_edge(FD(17)) then
        if DHT11_RESET='0' then--DHT11_driver 尚未啟動
            DHT11_RESET<='1';    --DHT11 資料讀取
            D7xx_xx<='0';        --:亮 (DHT11 資料讀取)
            times<=1400;        --設定計時
        elsif DHT11_ok='1' then--DHT11 讀取結束

```

```

        D7xx_xx<='1';          --:不亮 (DHT11 讀取結束)
        times<=times-1;        --計時
        if times=0 then        --時間到
            DHT11_RESET<='0';--DHT11 準備重新讀取資料
        end if;
    end if;
end if;
end process DHT11P_Main;

--蜂鳴器輸出
--濕度警報聲
HL<='0' when DHT11_DBoH>(conv_integer(SW8_2(7 downto 4))*10+
                        conv_integer(SW8_2(3 downto 0))) else '1';
sound1<=FD(22)and FD(16)and not HL;
--溫度警報聲
TL<='0' when DHT11_DBoT>(conv_integer(SW8_1(7 downto 4))*10+
                        conv_integer(SW8_1(3 downto 0))) else '1';
sound2<=not TL;

--DHT11 顯示
D7_data(0)<=DHT11_DBoH mod 10;    -- 濕度擷取個位數
D7_data(1)<=(DHT11_DBoH/10)mod 10; -- 濕度擷取十位數
D7_data(2)<=DHT11_DBoT mod 10;    -- 溫度擷取個位數
D7_data(3)<=(DHT11_DBoT/10)mod 10; -- 溫度擷取十位數

--4 位數掃描器--
scan_P:process (FD(17))
begin
    if rstP99='0' then
        scanP<=0;                --位數取值指標
        DHT11_scan<="1111";--掃描信號
    elsif rising_edge(FD(17)) then
        scanP<=scanP+1;
        DHT11_scan<=DHT11_scan rol 1;
        --DHT11_scan 必須為 unsigned
        --DHT11_scan<=DHT11_scan(2 downto 0) & DHT11_scan(3);
        --DHT11_scan 可為 unsigned 或 std_logic_vector
        if scanP=3 then
            scanP<=0;
            DHT11_scan<="1110";--掃描信號
        end if;
    end if;
end process scan_P;

--小數點控制 (閃爍表示超出設定)
with scanP select
    D7sp<=  HL when 0,--濕度
            HL when 1,--濕度
            TL when 2,--溫度
            TL when 3;--溫度

D7data<=(D7sp or FD(24)) & Disp7S;--七段顯示碼整合輸出

```

```

--BCD 碼解共陽極七段顯示碼 pgfedcba
with D7_data(scanP) select --取出顯示值
    Disp7S<=    "1000000" when 0,
                "1111001" when 1,
                "0100100" when 2,
                "0110000" when 3,
                "0011001" when 4,
                "0010010" when 5,
                "0000010" when 6,
                "1111000" when 7,
                "0000000" when 8,
                "0010000" when 9,
                "1111111" when others; --不顯示

--除頻器--
Freq_Div:process(gckP31)          --系統頻率 gckP31:50MHz
begin
    if rstP99='0' then           --系統重置
        FD<=(others=>'0');       --除頻器:歸零
    elsif rising_edge(gckP31) then --50MHz
        FD<=FD+1;                --除頻器:2 進制上數(+1)計數器
    end if;
end process Freq_Div;

end Albert;

```