

序

「CPLD 數位邏輯設計實務」是繼「8051 單晶片設計實務」之後的另一本「設計實務」系列書籍。「設計實務」系列書籍完全以學習者的角度來編寫，主要是想提供給學習者在程式設計上的實務經驗，以快速建立最正確的觀念，使學習者不但能有效提昇程式設計的能力，也能減少學習的時間及障礙。

本書是以美國 ALTERA 公司 MAX+PLUS II 軟體為發展工具，並以 VHDL 硬體描述語言為設計基礎，詳細說明數位電路設計的概念，內容涵蓋最完整的週邊介面應用，如 LED、矩陣型 LED、七段顯示器、LCD 模組、指撥開關、按鍵開關、直流馬達、步進馬達、伺服馬達、數位/類比轉換器(ADC)，類比/數位轉換器(DAC)、蜂鳴器、溫度感測器等。

本書內容編排深入淺出，由最簡單的數位邏輯—基本邏輯閘開始，引導學習者快速建立 VHDL 硬體描述語言的設計基礎，到最後能完成專題電路的製作。適合作為高職及大專院校電子、電機、資訊等專業相關課程的教材，對於從事電子工程設計的科技人員也是一本極佳的參考書。

本書強調數位邏輯電路的設計實務，藉由實作方式來介紹 VHDL 硬體描述語言的設計技巧，近 300 個實作範例程式，內容充實且應用生活化，學習者可以組合書中各單元的部份程式，輕鬆快速的完成專題電路製作。

本書為求內容完美、實用，已經多次研究與校正，但仍有疏忽之處，請讀者、專家、學者能給予指正，如要延伸學習者，歡迎至個人教學資源網。

楊明豐

99.10.1

教學資源網：<http://media.nihs.tp.edu.tw/user/yangmf/default.aspx>

商標聲明

- Altera 是 Altera 公司的註冊商標
- Max + Plus II 是 Altera 公司的註冊商標
- Xilinx 是 Xilinx 公司的註冊商標
- 除上述所列商標外，本書內文所提及的公司名稱及產品名稱，均為其所屬公司所有，特此聲明。

目錄

chapter 1 CPLD 簡介

1-1	何謂 CPLD.....	1-2
1-2	MAX+PLUS II 軟體下載與安裝	1-3
1-3	MAX+PLUS II 的 License 設定.....	1-10
1-4	安裝 Altera Byteblaster 燒錄驅動程式(Windows XP).....	1-15

chapter 2 圖形編輯設計

2-1	圖形編輯快速入門	2-2
2-1-1	建立新專案	2-2
2-1-2	開啟新檔案	2-3
2-1-3	繪製電路圖	2-4
2-1-4	檔案編譯	2-6
2-1-5	功能模擬	2-7
2-1-6	建立電路符號檔	2-13
2-1-7	晶片選擇與晶片腳位配置	2-14
2-1-8	程式下載	2-16
2-1-9	電路測試	2-17
2-2	圖形編輯的階層式設計	2-18
2-2-1	全加法器	2-18
2-2-2	全加法器實作練習	2-19
2-2-3	四位元並列加法器	2-25
2-2-4	四位元並列加法器實作練習	2-26

chapter 3 VHDL 語言基礎

3-1	前言	3-2
3-2	VHDL 基本架構	3-2
3-2-1	零件庫宣告	3-3
3-2-2	單體宣告	3-4
3-2-3	架構宣告	3-5
3-3	VHDL 資料型態	3-9
3-3-1	邏輯訊號	3-9

3-3-2 數值訊號.....	3-10
3-4 運算子	3-13
3-4-1 邏輯運算子	3-13
3-4-2 關係運算子	3-14
3-4-3 算術運算子	3-15
3-4-4 移位運算子	3-16
3-4-5 結合運算子	3-16
3-4-6 運算子優先順序.....	3-18
3-5 VHDL 資料物件	3-18
3-5-1 訊號(signal).....	3-18
3-5-2 變數(variable)	3-19
3-5-3 常數(constant)	3-20
3-6 VHDL 的敘述語法.....	3-20
3-6-1 共時性敘述(Concurrent Statements)	3-20
3-6-2 順序性敘述(Sequence Statements)	3-22

chapter 4 組合邏輯電路

4-1 基本邏輯閘.....	4-2
4-1-1 及閘	4-3
4-1-2 或閘	4-4
4-1-3 反閘	4-5
4-1-4 反及閘	4-6
4-1-5 反或閘	4-7
4-1-6 互斥或閘	4-8
4-1-7 互斥反或閘	4-9
4-2 組合邏輯	4-10
4-2-1 加法器(adder).....	4-10
4-2-2 減法器(subtractor).....	4-15
4-2-3 比較器	4-18
4-2-4 解碼器	4-20
4-2-5 編碼器	4-25
4-2-6 多工器	4-28
4-2-7 解多工器	4-30

chapter 5 順序邏輯電路

5-1 前言	5-2
--------------	-----

5-2	RS 門	5-2
5-3	正反器	5-3
	5-3-1 SR 正反器	5-3
	5-3-2 JK 型正反器	5-5
	5-3-3 D 型正反器	5-6
	5-3-4 T 型正反器	5-7
5-4	暫存器	5-8
	5-4-1 串入串出移位暫存器	5-9
	5-4-2 串入並出移位暫存器	5-10
	5-4-3 並入串出移位暫存器	5-10
	5-4-4 並入並出移位暫存器	5-11
5-5	計數器	5-11
	5-5-1 非同步計數器	5-12
	5-5-2 同步計數器	5-13
5-6	狀態機	5-16
	5-6-1 簡單狀態機	5-16
	5-6-2 Moore 狀態機	5-17
	5-6-3 Mealy 狀態機	5-17
	5-6-4 簡單狀態機設計	5-17
	5-6-5 Moore 狀態機設計	5-20
	5-6-6 Mealy 狀態機設計	5-22
5-7	計時脈波	5-24

chapter 6 LED 實習

6-1	相關知識	6-2
6-2	單燈右移	6-3
6-3	霹靂燈	6-6
6-4	交通號誌	6-9

chapter 7 按鍵控制實習

7-1	相關知識	7-2
	7-1-1 消除彈跳	7-2
	7-1-2 鍵盤掃描	7-3
7-2	指撥開關控制 LED 亮/暗實習	7-6
7-3	指撥開關控制 LED 閃爍速度實習	7-8
7-4	指撥開關控制 LED 右移速度實習	7-10

7-5	指撥開關控制 LED 變化實習	7-12
7-6	按鍵開關控制 LED 左、右移位實習	7-16
7-7	按鍵開關控制 LED 變化實習	7-19
7-8	矩陣鍵盤控制實習.....	7-24

chapter 8 七段顯示器實習

8-1	相關知識	8-2
	8-1-1 工作原理.....	8-2
	8-1-2 多工掃描.....	8-3
8-2	一位數 BCD 上數計數器	8-4
8-3	二位數 BCD 上數計數器	8-6
8-4	四位數 BCD 上數計數器	8-9
8-5	24 小時制電子鐘	8-12
8-6	按鍵開關控制一位數 BCD 計數器.....	8-16
8-7	按鍵開關控制二位數 BCD 計數器.....	8-20
8-8	按鍵開關控制四位數 BCD 計數器.....	8-24
8-9	矩陣鍵盤控制四位數七段顯示器	8-29

chapter 9 矩陣型 LED 控制實習

9-1	相關知識	9-2
	9-1-1 掃描原理.....	9-3
	9-1-2 移位原理.....	9-4
9-2	靜態字元顯示控制實習	9-7
9-3	靜態字串顯示控制實習	9-10
9-4	動態字元顯示控制實習	9-14
9-5	動態字串顯示控制實習	9-18
9-6	電梯顯示面板(一)控制實習	9-23
9-7	電梯顯示面板(二)控制實習	9-29

chapter 10 文字型 LCD 模組控制實習

10-1	相關知識	10-2
	10-1-1 LCM 接腳說明	10-2
	10-1-2 LCM 內部記憶體	10-4
	10-1-3 LCM 指令	10-7
	10-1-4 LCM 初始化	10-12
	10-1-5 LCM 讀寫週期	10-14

10-2	LCM 內建字型顯示控制實習	10-15
10-3	LCM 自建字型顯示控制實習	10-20
10-4	LCM 移位字型顯示控制實習	10-26
10-5	LCM 計時器實習	10-31
10-6	LCM 數字鐘實習	10-37

chapter 11 步進馬達控制實習

11-1	相關知識	11-2
11-2	步進馬達 1 相激磁正轉控制實習	11-6
11-3	步進馬達 1-2 相激磁正轉控制實習	11-8
11-4	步進馬達正轉、反轉控制實習	11-10
11-5	步進馬達正轉、反轉、加速、停止控制實習	11-13
11-6	步進馬達正轉、反轉、加速、減速控制實習	11-17
11-7	可設定步進數之步進馬達控制實習	11-22

chapter 12 伺服馬達控制實習

12-1	相關知識	12-2
12-2	固定角度型伺服機控制實習(一)	12-5
12-3	固定角度型伺服機控制實習(二)	12-8
12-4	連續旋轉型伺服機控制實習	12-13

chapter 13 直流馬達控制實習

13-1	相關知識	13-2
13-2	直流馬達轉速及轉向控制實習	13-9
13-3	智慧型風扇實習	13-11
13-4	單鍵控制智慧風扇實習	13-16
13-5	LED 調光器實習	13-21
13-6	拖尾效果霹靂燈實習	13-24

chapter 14 類比/數位(A/D)介面實習

14-1	相關知識	14-2
14-1-1	感測器	14-2
14-1-2	放大整形	14-3
14-1-3	A/D 轉換實習	14-3
14-2	A/D 轉換實習	14-9
14-3	0~5V 數位直流電壓表	14-13
14-4	0~50°C 數位溫度計	14-17

chapter 15 數位/類比(D/A)介面實習

15-1	相關知識	15-2
15-2	D/A 轉換實習	15-6
15-3	指撥開關設定 0~5V 電源供應器	15-10
15-4	鍵盤設定 0~5V 電源供應器	15-14

chapter 16 聲音控制實習

16-1	相關知識	16-2
16-2	單音產生器	16-4
16-3	救護車聲音產生器	16-7
16-4	電子琴	16-10
16-5	電子音樂盒	16-13

Appendix A CPLD 燒錄介面電路

A-1	ByteBlaster II 燒錄介面電路	A-2
A-2	44-Pin / 84-Pin 腳位圖	A-4

四、立即練習

1. 設計 8 位元 LED 控制電路，變化如圖 6-4(a)所示，間隔 1 秒。(ror2bit.vhd)
2. 設計一 8 位元 LED 控制電路，變化如圖 6-4(b)所示，間隔 1 秒。(rol1bit.vhd)
3. 設計一 8 位元 LED 控制電路，變化如圖 6-4(c)所示，間隔 1 秒。(rol2bit.vhd)

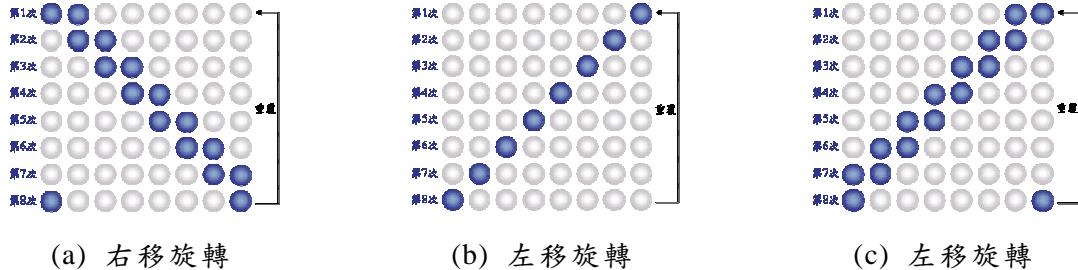


圖 6-4 LED 移位變化

6-3 霹靂燈

一、實習目的

設計一 8 位元 LED 控制電路，使其每 0.25 秒單燈 LED 左、右移位產生霹靂燈變化如圖 6-5 所示，每一個循環 16 種狀態。

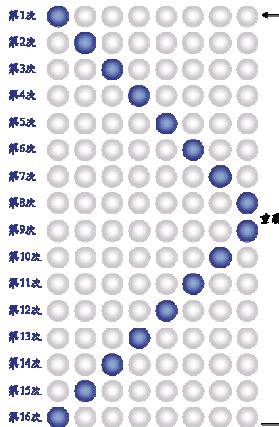


圖 6-5 LED 霹靂燈變化

二、功能說明

如圖 6-6 所示電路，除頻電路使用 SHIFT_CLK<=DIVIDER(13)，產生 $f = 32768 / 2^{13} = 4\text{Hz}$ (0.25 秒) 的移位脈波信號加至計數電路。因為是使用查表指令 with...select，依 LED 的變化，計數電路必須產生 00~15 計數值且在每個移位脈波信號的正緣上數加 1，因此定義計數值 COUNTER 為 0 到 15 的整數。解碼電路則依 COUNTER 值查表取出 LED 資料，LED 資料經反相後輸出至 LED 驅動電路。

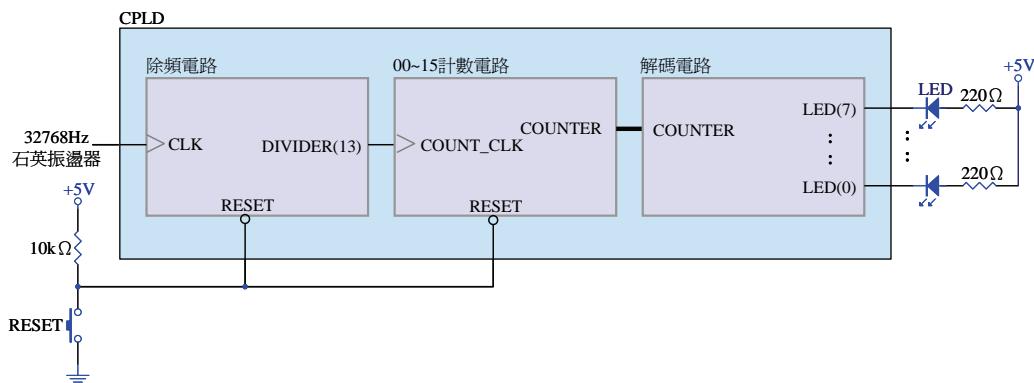


圖 6-6 霹靂燈電路方塊圖

圖 6-7

三、VHDL 程式(檔名:PILI1BIT.VHDL)

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity PILI1BIT is
    port
        (   RESET,CLK:in std_logic;           --輸入
            INV_OUT:out std_logic_vector(7 downto 0)  ); --輸出
end PILI1BIT;
architecture arch of PILI1BIT is
signal DIVIDER:std_logic_vector(15 downto 1);          --15 位元除頻器
signal SHIFT_CLK:std_logic;                            --移位脈波
signal SHIFT_OUT:std_logic_vector(7 downto 0);         --LED 資料
signal COUNTER:integer range 0 to 15;                 --0~15 計數器
begin
    --除頻電路
    process(CLK,RESET)
    begin
        if RESET='0' then
            DIVIDER<="0000000000000000";           --除頻器初值
        elsif CLK'event and CLK='1' then
            DIVIDER<=DIVIDER+1;
        end if;
    end process;
    SHIFT_CLK<=DIVIDER(13);                           --4Hz 移位脈波
    --計數電路
    process(SHIFT_CLK,RESET)
    begin
        if RESET='0' then
            COUNTER<=0;
        elsif SHIFT_CLK'event and SHIFT_CLK='1' then
            COUNTER<=COUNTER+1;                   --移位脈波正緣，--計數值加 1
        end if;
    end process;
    --解碼電路
    with COUNTER select
        SHIFT_OUT<="10000000" when 0,
                    "01000000" when 1,
                    "00100000" when 2,
                    "00010000" when 3,
                    "00001000" when 4,
                    "00000100" when 5,
                    "00000010" when 6,
                    "00000001" when 7,                  --依 COUNTER 查表
                                                --0 之 LED 資料
                                                --1 之 LED 資料
                                                --2 之 LED 資料
                                                --3 之 LED 資料
                                                --4 之 LED 資料
                                                --5 之 LED 資料
                                                --6 之 LED 資料
                                                --7 之 LED 資料

```

```
end process;  
--低音產生電路  
process(BEEP1,RESET)  
begin  
    if RESET='0' then  
        BEEP2<='0';  
    elsif BEEP1'event and BEEP1='1' then  
        BEEP2<=not BEEP2;           --BEEP2 頻率為 500Hz  
    end if;  
end process;  
--多工電路  
with NCLK select  
SOUNDOUT<=  BEEP1 when '0',      --NCLK 負半週輸出 BEEP1 信號  
                  BEEP2 when others;   --NCLK 正半週輸出 BEEP2 信號  
end arch;
```

四、立即練習

1. 利用 CPLD 產生電話聲音。
2. 利用 CPLD 產生警車聲音。

16-4 電子琴

一、實習目的

利用 CPLD 設計一電子琴電路，使用 8 個按鍵開關控制產生中音度 Do、Re，Mi、Fa、So、La、Si 及高音 Do 等八個音。

二、功能說明

如圖 16-6 所示電子琴實習方塊圖，除 $n/2$ 脈波產生電路產生頻率為 $2f_{OUT}$ 脈波信號 PULSE。除 2 方波產生電路將脈波信號能除 2 產生頻率為 f_{OUT} 的方波信號能 BEEP。移位暫存器由 TONE 控制，當 TONE=0 時，SOUNDOUT 輸出為 0，喇叭不產生聲音；當 TONE=1 時，SOUNDOUT 輸出頻率為 f_{OUT} 的方波信號能至喇叭驅動器，驅動喇叭產生聲音。按鍵電路偵測按鍵狀態，按下按鍵 1 時，TONE=32767/523/2 使喇叭產生 Do 音；按下按鍵 2 時，TONE=32767/587/2 使喇叭產生 Re 音；按下按鍵 3 時，TONE=32767/659/2 使喇叭產生 Mi 音；按下按鍵 4 時，TONE=32767/698/2 使喇叭產生 Fa 音；按下按鍵 5 時，TONE=32767/784/2 使喇叭產生 So 音；按下按鍵 6 時，TONE=32767/880/2 使喇叭產生 La 音；按下按鍵 7 時，TONE=32767/988/2 使喇叭產生 Si 音；按下按鍵 8 時，TONE=32767/1046/2 使喇叭產生高音 Do 音。

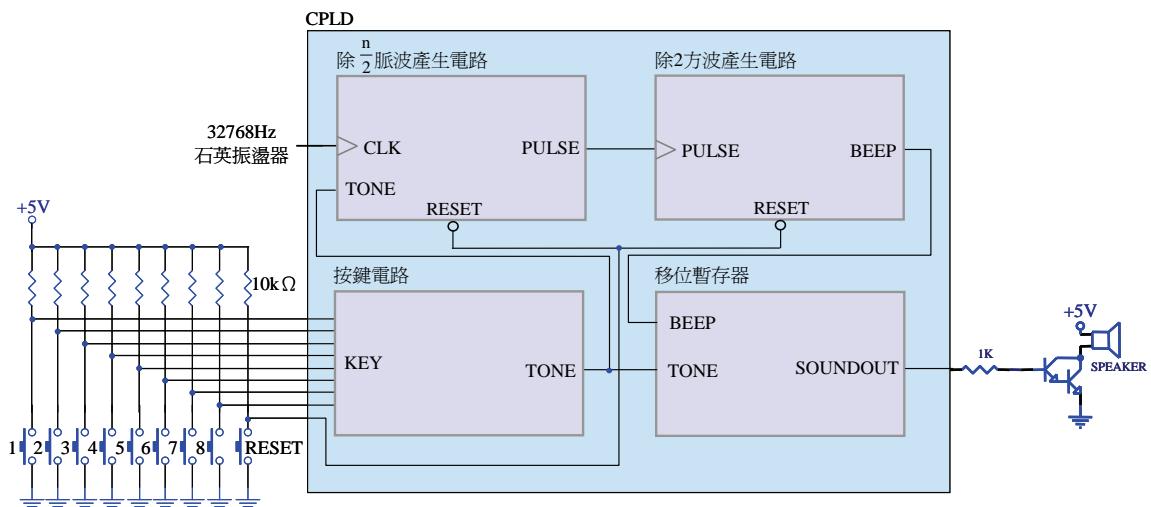


圖 16-6 電子琴

三、VHDL 程式(PIANO.VHD)

```

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity PIANO is
    port
        (   RESET:in std_logic;           --重置輸入
            CLK:in std_logic;           --32768Hz 計時脈波輸入
            KEY:in std_logic_vector(7 downto 0); --8 位元按鍵輸入
            SOUNDOUT:out std_logic );  --聲音輸出
end PIANO;
architecture arch of PIANO is
signal PULSE:std_logic;           --除 n/2(2fOUT)脈波信號
signal BEEP:std_logic;           --除 n(fOUT)方波信號
signal COUNT:integer;             --音階除頻常數
signal TONE:integer;
begin
--除 n/2 脈波產生電路
process(CLK,RESET)
begin
    case KEY is
        when "11111110"=>TONE<=32767/523/2;      --C(Do:523Hz)
        when "11111101"=>TONE<=32767/587/2;      --D(Re:587Hz)
        when "11111011"=>TONE<=32767/659/2;      --E(Mi:659Hz)
        when "11110111"=>TONE<=32767/698/2;      --F(Fa:698Hz)
        when "11101111"=>TONE<=32767/784/2;      --G(So:784Hz)
        when "11011111"=>TONE<=32767/880/2;      --A(La:880Hz)
        when "10111111"=>TONE<=32767/988/2;      --B(Si:988Hz)
        when "01111111"=>TONE<=32767/1046/2;     --C(Do:1046Hz)
        when others      =>TONE<=0;
    end case;
    if RESET='0' then
        PULSE<='0';
    elsif CLK'event and CLK='1' then
        if TONE=0 then
            COUNT<=0;                                --未按任何鍵，靜音輸出
        else
            COUNT<=COUNT+1;
            if COUNT=TONE then
                PULSE<='1';
                COUNT<=0;
            else
                PULSE<='0';
            end if;
        end if;
    end if;
end process;
end arch;

```