

以 FPGA 實現一個類 PIC 8 位元微控制器

黃登淵 柯憲棠

大葉大學電機工程學系

彰化縣大村鄉山腳路 112 號

摘要

本研究主要是以 VHDL (very high speed integrated circuit hardware description language) 硬體描述語言之晶片設計方法，來實現一個類 PIC16F877 之 8 位元微控制器 (microcontroller)。根據 PIC16F877 之 RISC (reduced instruction set computer) 架構與指令功能，採用 Altera 提供之實驗平台 UP2 來進行設計，設計完成之後，再經由 FPGA (field programmable gate array) 之實際驗證，結果顯示所設計之 FPGA 晶片已達原先所規劃之目標。

關鍵詞：Microcontroller, PIC, FPGA

FPGA Implementation of a PIC-like RISC Controller

DENG-YUAN HUANG and HSIEN-TANG KE

Department of Electrical Engineering, Da-Yeh University

112 Shan-Jiau Rd., Da-Tsuen, Changhua, Taiwan

ABSTRACT

This study sought to implement an eight-bit PIC16F877-like microcontroller by using VHDL (very high speed integrated circuit hardware description language) hardware language, which is extensively used in the chip-design industries. In this research, an Altera UP2 design package is employed to implement a similar RISC (reduced instruction set computer) architecture and instruction set to the PIC16F877. After completing the VHDL design, it is then repeatedly programmed into the FPGA (field programmable gate array) laboratory package to verify whether the functions have been fully achieved. The results indicate that we have already successfully implemented an eight-bit PIC16F877-like microcontroller.

Key Words: microcontroller, PIC, FPGA

一、前言

微控制器在日常生活中的應用可說是越來越普遍，其應用範圍從家電、汽車、防盜、電視遊樂器、電腦周邊產品，一直到通訊產品等，已逐漸地取代了傳統的機械式控制，微控制器不但可以降低成本，更可以增加產品之附加價值。由於其應用領域不斷擴大，所以大部分的產品都有嵌入式的微控制器，並可依特定功能來撰寫所需之應用程式，因此儘管電子產業發展迅速，但微控制器所佔的份量卻是有增無減。

FPGA (field programmable gate array) 常被使用於設計微控制器，例如：設計一顆 Pipeline stage CPU 相容於 8051 複雜指令集 (complex instruction set computer, CISC) [1]。此外，也以 FPGA 實現 Motorola MC68HC05 兩組 8 位元平行輸出 / 入埠及一組 UART (universal asynchronous receiver transmitter)，通常作為串列傳輸的介面 [7]。另有 8 位元 RISC 微控制器 AT90S1200 [5]，以及 32 位元 RISC 微控制器 MIPS (micro processor without inter locked pipe line stoges) 架構 [2] 和 8 位元 RISC 微控制器 PIC16C65 [3]，同樣以 FPGA 實現且被驗證成功。

本研究設計的 RISC (reduced instruction set computer) 微控制器可應用於各類指令集，不需建構大量的資料庫，只需在主程式中寫出一個通用型的 RISC 微控制器架構，再將各個管線階段的控制訊號產生出來，然後再呼叫副程式所需之 VHDL (very high speed integrated circuit hardware description language) 程式碼即可；應用此種方法，雖然設計出來的面積較大，但其重覆使用性較好，應用範圍也較為廣泛，並可依照使用者需求重覆設計，所以並不只侷限於某一種功能。

二、微控制器基本架構

(一) 中央處理單元 (CPU)

從記憶體讀入指令並執行，即負責整個微控制器的運作。包括三個子單元：指令解碼與控制單元、算術邏輯單元、程式計數器。

1. 指令解碼與控制單元：任何程式指令的運算碼 (opcode) 都先從程式記憶體讀入指令暫存器中，加以解碼再透過控制單元 (control unit) 發出時序訊號，使資料作適當的傳送與運算。
2. 算術邏輯單元 (arithmetic logic unit, ALU)：負責執行算術運算及邏輯運算的部份，通常 ALU 的輸入是累加器

(accumulator，簡稱 ACC 或 A 暫存器) 及臨時暫存器 (temporary register, TMP)，運算結果則送回累積器或透過匯流排送至記憶體或 I/O port。

3. 程式計數器 (PC)：計數器會自動指出存於程式記憶體中下一個待執行指令的存放位址，以便程式控制單元去讀取。

(二) 記憶體 (memory)

用來儲存程式及資料，常用的有 ROM、EPROM、RAM。

1. 程式記憶體：主要用途是儲存程式 (program)。在 PIC16F877 中，有 8K x 14 words 的快閃記憶體 (FLASH program memory)。
2. 資料記憶體：RAM 的主要用途是在程式運作中，暫存器存放資料的地方。在 PIC16F877 中，有 368 x 8 bytes 的資料記憶體 (RAM) 和 256 x 8 bytes 的資料記憶體 (EEPROM)。

(三) 輸入 / 輸出埠 (I/O port)

輸入埠負責將外界命令或資料載入微控制器。輸出埠負責將微控制器處理結果送至外界。目前很多控制器所用元件既可當輸入埠也可以當輸出埠使用，因此常合稱為 I/O port。

三、8 位元 PIC 微控制器規格及指令集

本研究實現的 IC 為 Microchip PIC16F877 微控制器，其規格 [6] 如下：

1. 八位元 RISC 微控制器。
2. 指令集共有 35 個指令，如表 1 所示。
3. 大部分為單一指令週期 (single cycle)，少部份分支指令 (program branch) 為 2 個指令週期。
4. 程式記憶體 (program memory, Flash ROM) 8Kx14 words。
5. 資料記憶體 (data memory)：368x8 bytes (RAM)，256x8 bytes (EEPROM)。
6. 八階層 (level) 堆疊 (stack)。
7. 五組輸入 / 輸出埠 (I/O port)：PORTA (6-bit)，PORTB~PORTD (8-bit)，PORTE (3-bit) 共 33-bit。
8. 三個計數器 (timer)。
9. 看門狗計時器 (watchdog timer)。

表 1. 14-bits core 指令集

位元組操作指令		常數操作及控制指令	
NOP	- No Operation	SLEEP	- Standby mode
MOVWF	f Move W to f	CLRWDT	- Clear watchdog
CLRWF	- Clear W	RETLW	k Ret., W=literal
CLRF	f Clear f	RETFIE	- Ret. from interrupt
SUBWF	f,d f-W	RETURN	- Ret. from subroutine
DECF	f,d Decrement f	CALL	k Call subroutine
IORWF	f,d W OR f	GOTO	k Go to address k
ANDWF	f,d W AND f	MOVLW	k Move literal to W
XORWF	f,d W XOR f	IORLW	k Literal OR W
ADDWF	f,d W+f	ADDLW	k Literal + W
MOVF	f,d Move f	SUBLW	k Literal - W
COMF	f,d Complement f	ANDLW	k Literal AND W
INCF	f,d Increment f	XORLW	k Literal XOR W
DECFSZ	f,d Dec. f, skip if 0		
RRF	f,d Rotate right		
RLF	f,d Rotate left		
SWAPF	f,d Swap Nibble		
INCFSZ	f,d Inc. f, skip if 0		
BCF	f,b Bit clear f		
BSF	f,b Bit set f		
BTFSZ	f,b Bit test, skip if 0		
BTFSZ	f,b Bit test, skip if 1		

註：f=暫存器或記憶位址，k=常數值（8-bit），b=第幾位元<0,7>，d=運算後資料目的地（0=f, 1=W）

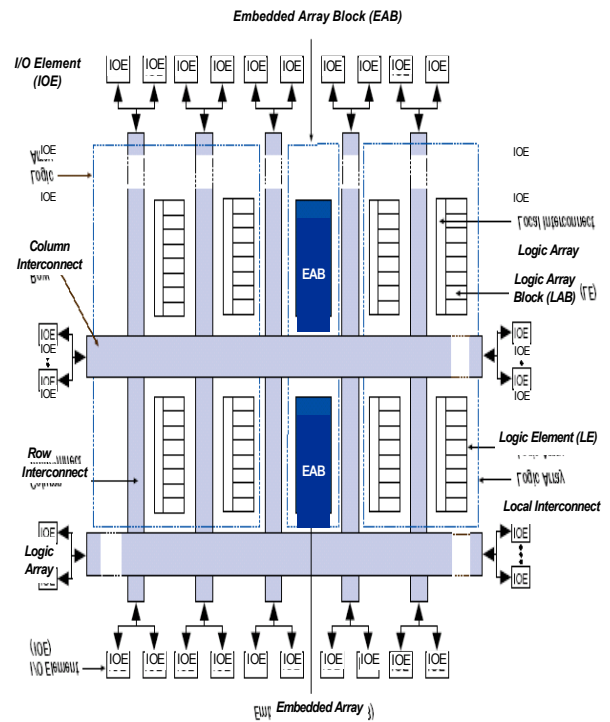


圖 1. FLEX10K 方塊圖

四、8 位元 RISC 微控制器設計

本研究數位電路係採用 Altera 公司的 FLEX10K70 [4] 系列 FPGA 元件來進行數位電路整合設計。此系列為 SRAM-Based LUT (look up table) Devices，即內部以 SRAM 與查表的方式來規劃 FPGA 元件，FLEX10K70 詳細規格與特色如表 2 所示，FLEX10K 元件內部如圖 1 所示，可實現記憶體及特定邏輯函數的內嵌陣列 (embedded array)，並可實現一般邏輯陣列 (logic array)。

本研究設計 8 位元 RISC 微控制器首先由建表規劃開始，並輸入所需要產生微控制器的規格和指令集；輸入建表中包含資料記憶體、輸入 / 輸出埠 (I/O port)、特殊使用暫存 (SFR)、算術邏輯單元 (arithmetic logic unit)、指令集、

表 2. FLEX10K70 系列特色

Typical gates (logic and RAM)	70,000
Maximum system gates	118,000
LEs	3,744
LABs	468
EABs	9
Total RAM bits	18,432
Maximum user I/O pins	358

堆疊大小和用來產生計數器 / 看門狗計時器所需要的位元大小。

(一) 8 位元 RISC 微控制器之設計流程

圖 2 為 8 位元 RISC 微控制器之設計流程，其可分為三部份：

1. 收集欲設計之微控制器相關資訊，如：指令集的定義、特殊功能的使用、資料記憶體的規劃等。在建表時，並將所需要的資料逐一填入表格中。

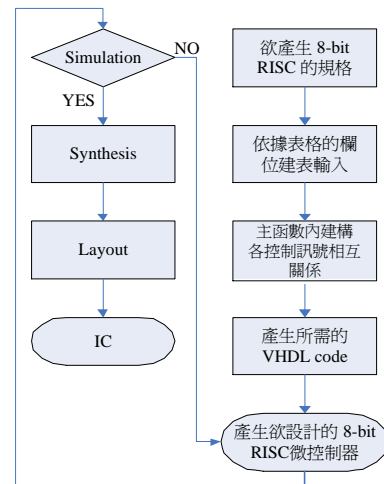


圖 2. 8-bit RISC 微控制器之設計流程

2. 將建表後得到的完整資料，產生所要的 8-bit RISC 微控制器的 VHDL 硬體描述語言。
3. 產生 VHDL code 後，先以時序模擬 (timing simulation) 的方式來進行驗證，是否符合所要設計之規劃。

(二) 四級管線之設計

此一 8 位元微控制器採用四級的管線 (pipeline 4 stage) 設計，根據圖 3 之 8 位元 RISC 微處理器通用型架構，再依照圖 4 之流程設計圖來設計四級管線，其方法如下：

1. 擷取 (fetch) 階段：微處理器擷取指令並將其存至指令暫存器 (instruction register, IR)，以供下一級讀取，且程式計數器 (PC) 自動加一以指向下一個指令位址。
2. 解碼 (decode) 階段：從指令暫存器中取出二元碼 (binary code) 資料，並將來源端 (source) 與目的地端 (destination) 之控制訊號解碼出來，同時解碼出所需之運算元與運算結果欲儲存之有效位址。
3. 執行 (execute) 階段：由解碼階段取出來源端所在位址之資料，或由指令暫存器取得所需之常數資料。此外，算術邏輯單元也同時解碼出運算模式，再將運算結果存入至「輸出緩衝暫存器」內，且在此階段中也產生特殊暫存器所需之寫入控制訊號。
4. 寫入 (write back) 階段：將邏輯運算單元所計算之結果，存入已產生寫入控制訊號之特殊暫存器內。

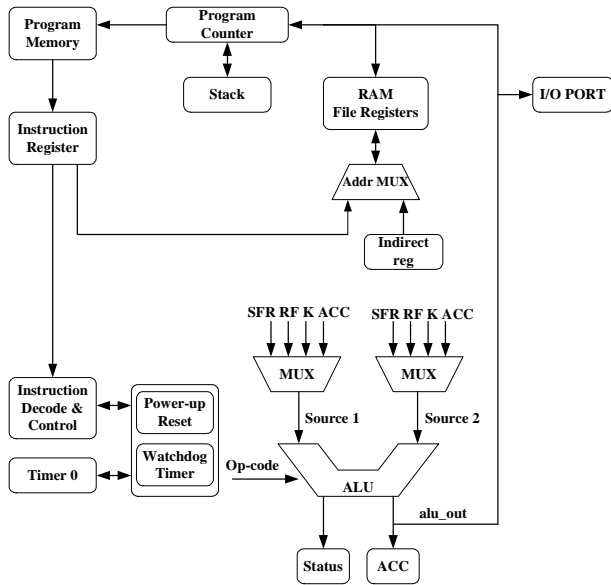


圖 3.8 位元 RISC 微處理器架構

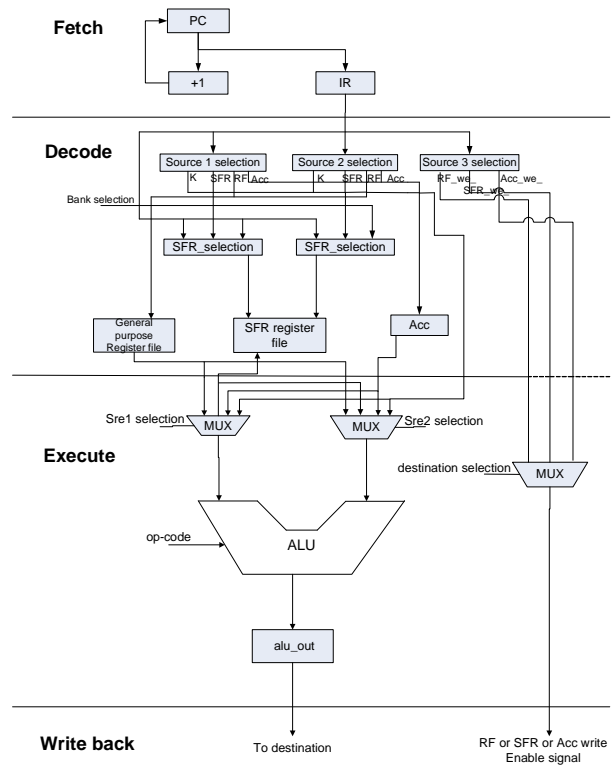


圖 4.8 位元 RISC 微處理器之管線設計流程

(三) 建表的規劃及使用

建表規劃所需之資料包括：資料記憶體的使用定義、基本輸入 / 輸出埠、特殊暫存器的使用、算術邏輯單元、指令集及特殊功能的定義。其中又以指令集的定義最為複雜，圖 5 為建表的種類及說明。

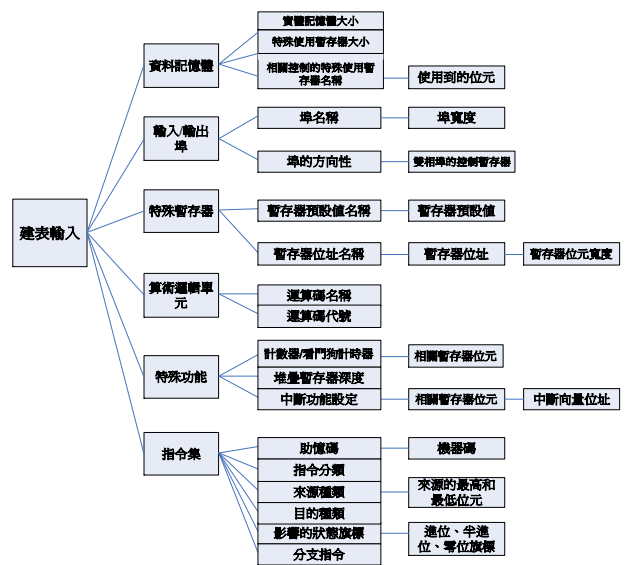


圖 5. 建表種類說明

五、FPGA 功能之驗證

8 位元 RISC 微控制器經時序模擬驗證無誤後，接著進行 FPGA 模擬板之驗證工作，圖 6 為 FPGA 驗證之流程，首先將 VHDL code 載入至 Max-plus II 軟體內以合成 (synthesis) 產生出 *.vhd 檔，再經由編譯 (compile) 產生 *.sof 檔，其中 *.sof 檔為可燒錄檔，其可用來載入至 FPGA 內部以進行驗證工作，若結果與原先規劃不符，則重複修改 VHDL code 至符合原先規劃之目標為止。

為了驗證 VHDL 程式是否符合原先規劃之功能，本文選用兩位數之計數範例以測試功能是否達成：

1. 應用電路：開關電路、七段顯示電路、LED 電路，以上電路是為了測試應用程式功能所設計之電路。
2. 功能：從 00~99 計數，每 0.1 秒計數一次。
3. 測試目的：測試基本指令、跳躍指令、程序呼叫指令和埠的輸入 / 輸出是否正常。

經由圖 7 之 FPGA 外部電路顯示，已達到原先所規劃之目標。

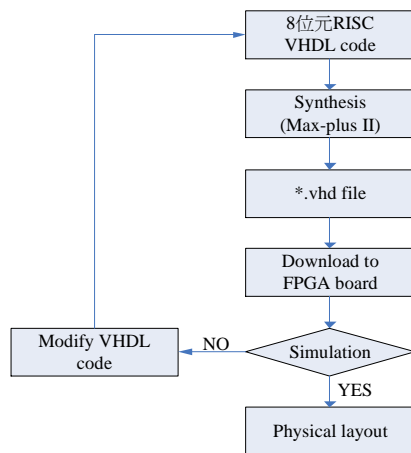


圖 6. FPGA 驗證之流程圖



圖 7. 電路驗證圖

六、結論

FPGA 可重複修改與燒錄之特性，使它成為驗證積體電路設計之有力工具；近年來，隨著 FPGA 閘數 (gate count) 之不斷提高與單位閘級成本之不斷下降，已使得它被廣泛地應用在 IC 設計之各個領域上，特別是在 ASIC 方面之開發，工程師藉由它可迅速產生 IC 元件雛形之特性，以縮短產品之上市時間。

綜合以上結果，經由 FPGA 模擬驗證本研究類 PIC 8 位元微控制器之設計為正確可行，在未來更可延伸至 16、32 位元 RISC 微控制器 VHDL code 之產生，甚至是 DSP 控制器之實現。

參考文獻

1. 張哲彰 (民 83)，管線化 8 位元微控制器，國立成功大學電機工程研究所碩士論文。
2. 劉俊佑 (民 83)，發展運用似 MIPS 架構之微控制器，大葉大學電機工程研究所碩士論文。
3. 賴瑞明 (民 82)，自動產生 8-bit RISC 微控制器硬體描述語言之設計，南台科技大學電機工程研究所碩士論文。
4. Hamblen, J. O. and M. D. Furman (2002) *Rapid prototyping of digital systems. A Tutorial Approach*, 2nd Ed., Kluwer Academic Publishers, Boston.
5. He, Y. Z. (2002) *Building A RISC Microcontroller in an FPGA*, M.S. thesis, Department of Electrical Engineering, universiti Teknologi Malaysia.
6. Microchip (2001) datasheet, <http://www.microchip.com.tw>.
7. Roth, C. H. (1998) *Digital System Design using VHDL*, PWS Publishing Company, Boston.

收件：94.02.21 修正：94.04.13 接受：94.06.09