

16位元微處理機系統 結構、程式設計及應用

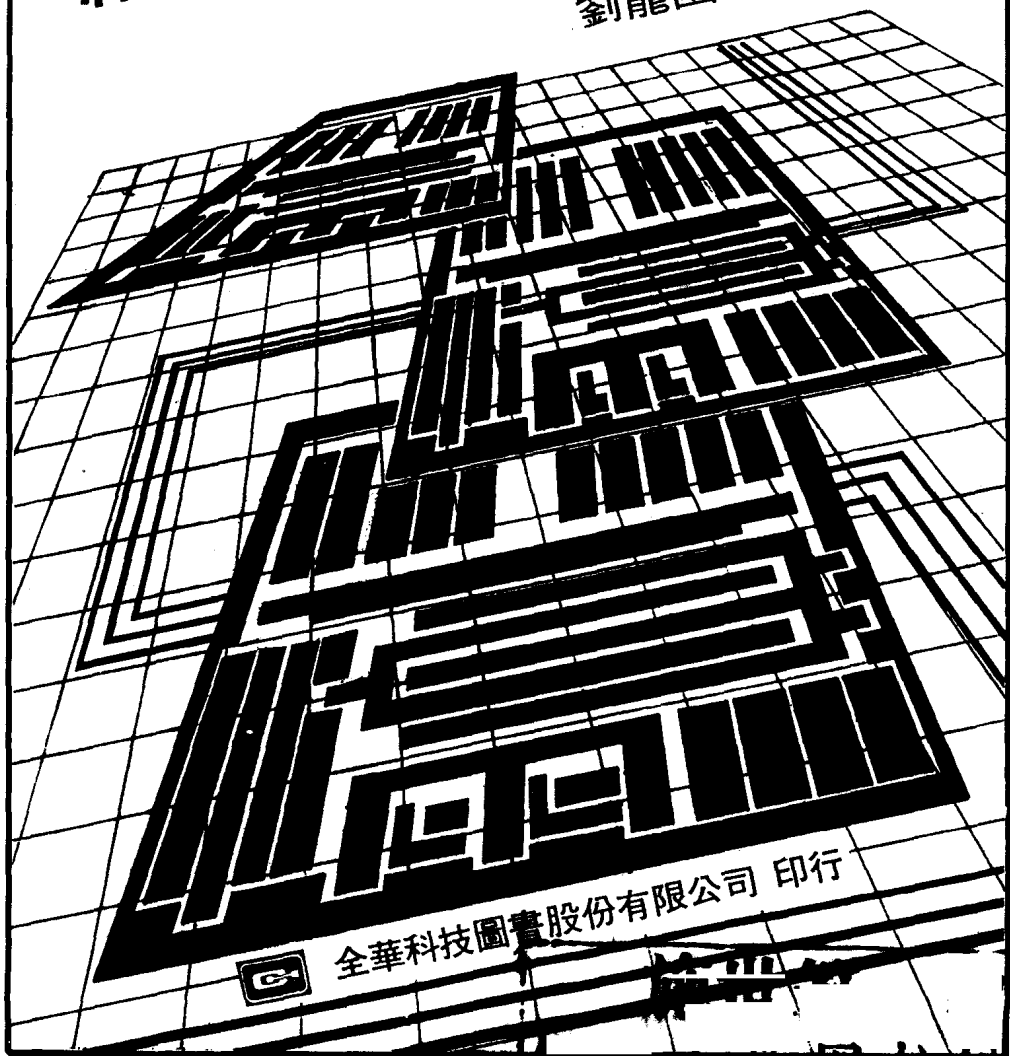
劉龍國 編譯



 全華科技圖書股份有限公司 印行

16位元微處理機系統 結構、程式設計及應用

劉龍國 編譯



全華科技圖書股份有限公司 印行



全華圖書

法律顧問：陳培豪律師

16位元微處理機系統 結構、程式設計及應用

劉龍國 編譯

出版者 全華科技圖書股份有限公司
地址 / 台北市龍江路76巷20-2號2樓
電話 / 5811300 (總機)
郵撥帳號 / 0100836 - 1 號

發行人 陳 本 源
印刷者 華 一 彩 色 印 刷 廠

門市部 全友書局(黎明文化大樓七樓)
地址 / 台北市重慶南路一段49號7樓
電話 / 3612532 • 3612534

定 價 新臺幣 **320 元**
二版 / 76年 8 月

行政院新聞局核准登記證局版台業字第〇二二三號

版權所有 翻印必究

圖書編號 0121309

MICROPROCESSOR SYSTEMS

A 16-Bit Approach

William J. Eccles

University of South Carolina

HB20/09

原序

本書 16 位元微處理機系統結構、程式設計及應用將帶領我們對微處理機系統的結構、程式設計及應用作一次豐富之旅。它主要的目的乃是做為在微處理機方面以程式設計為主要重點的第一門課程。同時也可以明確地選取本書中的部份題材做為工程或計算機科學方面的課程。

本書有三個主要的目標：第一，我希望使用非脅迫的方式而以組合語言的層次來介紹程式的設計，但同時要保有程式設計所應有的品質及結構。第二，我希望對 16 位元的處理機做一番介紹，因為它們有功能強的特性而又不至於太複雜，因為它們受大眾的喜愛而頗為流行，更因為它們使用起來頗為有趣。第三，我希望以迎合大學生口味的語調來寫這本書，因為這將有助於學生在充實其基礎的同時能喜愛這些題材。

這些目標大部份是以例題的型式來達成，我們以一個一般的設計問題做為開始，然後介紹微處理機，最後再處理許多不同的細節工作。本書從頭到尾均是以 Motorola 的 MC 68000 做為主要的處理機，並在許多地方亦使用到 Intel 的 iAPX 86/10 (8086)。我之所以選用 68000 乃是因為我比較喜歡這個處理機，同時也覺得這個處理機比較好設計。而包含了 8086 的介紹則可提供做比較之用，並說明了對於同一個需求可以使用各種不同的方式來達成，同時也可以讓學生們感受到賣主有著各種不同的構想存在。本書於結束時再度討論第一章中所介紹的一個一般性的設計問題，並使用 68000 來完成該系統之設計。

本書之特色

1. 本書以一個微處理機系統設計的例題做為開始，並以完成該系統之設計做為結束，完整的列出零件的明細表。
2. 書中包含有許多的例題，而且每一個論題均不失其明確性、結構或品質。
3. 每次需要使用高階語言來闡明組合語言的解答時，均是使用AdaTM 程式規劃語言。〔AdaTM 乃是美洲政府的註冊商標（Ada 聯合計劃中心）。〕
4. 本書從頭到尾均使用Motorola的MC 68000 做為主要的處理機，故幾乎包含了這個處理機所有程式規劃的特色。
5. 書中亦包含了 Intel的iAPX 86/10(8086) 做為第二個處理機以做比較。
6. 本書中包含有硬體的設計以說明其應用情形，並提供有程式規劃的例題。但僅使用標準的介面。

必備條件

使用本書做為課本的學生必須具有兩個領域的基礎。雖然我們所使用的僅是初級電機工程的水準，但本書亦適用於一知半解的學生。學生必須要有一些高階語言之計算機程式規劃的經驗。PASCAL 及結構化的FORTRAN 便是一項很好的選擇，假如能夠忘掉不好的習慣，則BASIC 可能會更好些。學生們也應該要有一些數位邏輯的基礎。不過這並不是絕對必要的，假如你是使用本書而又沒有這項基礎時，只要跳過這些電氣的題材即可。

內容及組織

第一章及第二章是介紹性的內容。第一章以一個百合花控制器的設計問題做為開始，並討論了設計程序的一般方法。這一部份的題材可以做為簡單的起頭或指定閱讀使用。第二章則是對計算機、微處理機，及本課程所需之字彙做一般性的介紹。雖然大多數的賣主均有其自己的術語，但熟悉標準的字彙却是必要的。

第三、四、五章則是介紹兩個處理機。其中第三章較其他兩章來得長，這是因為在第三章中不僅是介紹MC 68000而已，同時亦討論了二進數及有條件式運算的一般題材。第四章則包含了對8086題材之討論。在這兩章結束之後，同學們應該已經遇到了這兩種處理機之順序及條件結構。第五章則發展出這兩個處理機的重覆結構，至於每個處理機所有之特殊的迴路控制指令則留到後面討論。

第六章是有關輸入及輸出的技巧，其中包含有程式控制I/O及中斷驅動I/O，而主要的重點則是放在68000及中斷驅動I/O上面。我覺得當學生能夠寫程式時，中斷乃是一項值得介紹的重要觀念。同時它亦提供了一個很好的例題來源。在本章結束時亦討論了8086對中斷處理的方式以做為與Motorola之技巧的比較。

在第七章中則強調了程式分段的必要性及介紹了副常式的觀念。我嚐試著去闡明一個大的程式必須要分解成許多小部份的觀念。在其同時，我也強調了要有一個好的技巧來建立這些程式以使得能產生可工作的軟體。我們使用了許多實質上的例題來說明這些原則，而我覺得也只有利用例題才能有效的說明這些原則。本章最後則是以選讀的68000連結及解連結指令做為結束，這兩條指令對於副常式的連結提供了功能強大的工具。我們亦使用了漢諾宜塔來介紹這個論題及遞迴。

第八、九章則是處理輸入及輸出的介面，在這兩章中均使用標準的介面。在第八章中使用了Motorola的週邊介面配接卡(MC 6821)，並完全使用68000做程式規劃。在第九章中則使用Intel的通用同步——非同步接收——傳送器(8251A)，但僅討論了其中不同步的模式。而程式規劃則完全使用8086的指令。在這兩章中我們簡要地討論到處理機匯流排的結構，對匯流排的介面、介面裝置本身，以及一些這些裝置的應用例題。而沒有任何一章是包括了所有的部份——這個

論題應該是在其他的課程中討論。(在第九章中以8086取代68000做為主要的處理機來討論，以便在第十章的某些地方對其特色做更深入的討論。)

在第十章中收集了許多例題，藉以說明一些額外的程式規劃技巧以及一些有趣而普遍的資料結構。兩個處理機均有程式用以說明位元之處理及BCD的算術。字串的處理則以8086的指令為主，而68000的例題則較少。至於浮點運算則完全使用68000的指令。包含陣列的例題則大部份均使用68000。本章的主要論題乃是獨立的。

第十一章用來完成第一章中所介紹之百合花控制器之設計。雖然這項設計使用68000(及Motorola的教育計算機板)是有點大材小用，但我却覺得這是正當的。因為我希望能因此有一個相當簡單的例題，而不至於花掉太多的時間及空間。此項設計乃是根據硬體設計及程式規劃的規格來完成的。其中包括了所有的程式及電路。

使用法

到底你能涵蓋本書多少內容呢？對於一個學期來講，本書的內容是太多了一點，所以你應該要選取一些能夠迎合你及你的學生口味的論題。例如，雖然第四、五章中8086的題材或許不能跳過，但你却可以根據你的需要選取8086之題材。

68000的連結及解連結便可省略。第八、九章中的介面則可縮短或省略之。第十章則可選取你所需要之題材。第十一章則可完全略去或為指定閱讀，也可以用來做為計劃或實驗的習題。(我曾使用過各種組合方式做為一個學期的課程，但其中我較喜歡使用從第一章到第八章的所有題材及第十章的大部份)。

我寫這本書的主要方式乃是希望能夠適合學生們的閱讀，而不在於迎合選用本書為課本的教授們。因此，我儘量的使用口語化的方式來介紹，希望不會對學生們的學習產生太大的困擾。例如在本書中有許多地方，當我們遭遇到困難時，我就會

使用一些俚語來輕鬆一下。當然，你也可以從本序言中看出，我並不會以高姿態的方式來解說任何一件事。我的學生們告訴我，我寫的這一本書就像講課一樣地輕鬆。這種表達方式適用於我和我的學生之間，我相信也一定能夠適用於你們才對。

這些教材以這種安排方式使用已經有兩年的歷史了。對於電機工程的大三學生需要有一學期每週四小時的課程，並包括以 68000 為基底的實驗室。我並不要求學生們去熟記指令集，而對於所有的小考均是採用可看書的方式來進行。因為我發現到常常做短時間的小考（一星期大約 20 分鐘）其效果將比只考幾次的期考來得好。雖然我也指定了每章後面的問題做為家庭作業，但這些家庭作業是不計成績的。因為我會在課堂上回答學生們所提的任何問題，而學生們也都知道這些小考就可反映出學生對家庭作業認識的程度。

譯者序

目前，微處理機及微算機已普遍的使用於工業控制系統、資料處理、程序控制、交通控制、科學儀器、軍事設備，及電視遊樂器等方面，故微處理機與微算機，不僅給電子界帶來無比的震撼力量，同時也造成了家用電器的革命，改進了人類的生活品質。在電子工程領域裏，微處理機取代了傳統的數位系統及邏輯設計；在計算機科學領域裏，微處理機則成為迷你計算機與傳統計算機的勁敵。尤其是 16 位元的微處理機，已經具有許多大型計算機才有的功能。因此，瞭解微處理機與微算機系統的工作原理，乃至於應用、設計是絕對必要的。

今日，專注於有關微處理機和微算機電子系統設計的人們，必須具有三個主要領域的完整知識：微處理機系統的結構、微處理機的軟體，以及硬體介面技術。然而針對這些主題，目前街坊間可取得的大部份書本，均著重於微處理機的系統結構、指令集，和程式設計的技巧。一般而言，甚少的資訊提供在硬體設計和介面技術上，而這些資訊則是微處理機系統能否成功地應用之關鍵。

本書 16 位元微處理機系統結構、程式設計及應用則包括了前述三個主要領域的知識。它提供了我們對微處理機系統的結構、程式設計及應用，做了一次豐富之旅。其中我們以一個一般的設計問題做為本書的開始，然後介紹微處理機系統的結構、微處理機的軟體，以及硬體介面技術，最後再以一個完整的系統設計完成該設計問題做為結束。在這種前後呼應的安排方式之下，能夠使讀者對於一個設計問題從頭到尾的整個過程有所認識。

學習微處理機及微算機，必須由認識微處理機的內部結構及動作原理開始。然後，學習如何應用該處理機的指令集來寫程式。進而再熟悉其輸入／輸出介面之構造、功能，及程式規劃等。爲了讓讀者能夠順利地完成這整個學習過程，同時能夠獲取清楚的概念，所以，本書大部份均是以例題的方式來介紹。其中則是以Motorola的MC 68000 做爲主要的處理機，但在許多地方亦會使用到 Intel 的 iAPX 86/10 (8086)微處理機，以便能夠與MC 68000 做一比較。

由於Motorola的MC 68000 以及 Intel 的 iAPX 86/10 (8086)乃是目前最流行的兩個 16 位元的微處理機，而本書則以這兩個 16 位元的微處理機做爲介紹的範疇，其中幾乎包含了MC 68000 的所有特性。再加上書中對實際設計問題的探討，提供了硬體的設計技術及軟體程式規劃的例題，而使得這本書成爲一本相當實際、有用的好書。

譯者在此要感謝我的朋友鍾淑欽小姐以及其他的一些朋友，他們幫我閱讀原稿、做資料的搜集與校訂，並由於他們的建議，使本書有重大的改善；我們亦得感謝全華圖書公司所有同仁的協助與支持，由於他們幫助，使得本書順利出版。特予誌謝。

本書之編譯，乃公餘之暇匆促成書，雖經譯者盡力，不免仍有疏漏謬誤之處，尚祈讀者先進包涵，並不吝批評指教！

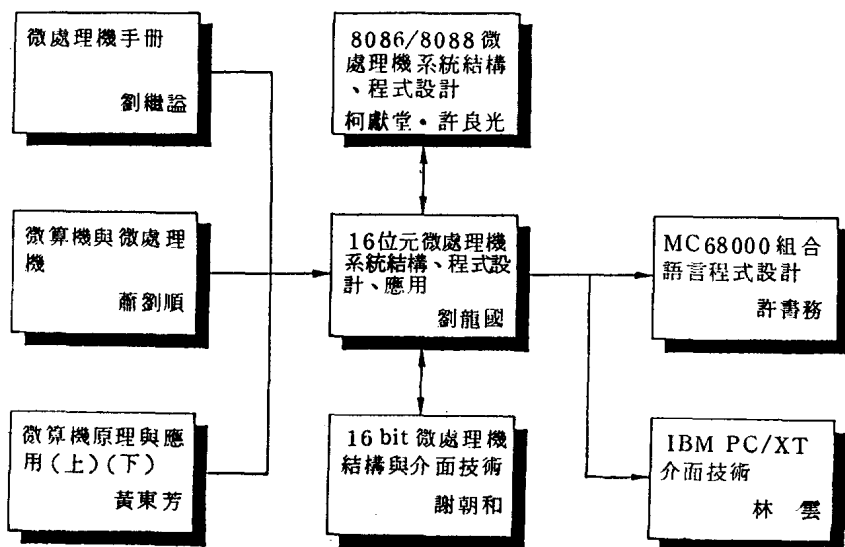
劉龍國 謹誌

編輯部序

「系統編輯」是我們的編輯方針，我們所提供給您的，絕不只是一本書，而是關於這門學問的所有知識，它們由淺入深，循序漸進。

本書涵蓋了微處理機系統結構、軟體及介面技術三大部份，並以例題的方式做為引導，使讀者在例題的解說下更能了解及融會貫通。本書概分十一章，敘述對象為 MC 68000 及 8086，例題多且詳細，並且連貫系統、軟體、介面三個單元，讀者在循序漸進的研讀下必對這兩種微處理機有一精詳的理念，故為大專電子科系計算機結構最佳教材。

同時，為了使您能有系統且循序漸進研習微處理機方面叢書，我們以流程圖方式，列出各有關圖書的閱讀順序，以減少您研習此門學問的摸索時間，並能對這門學問有完整的知識。若您在這方面有任何問題，歡迎來函連繫，我們將竭誠為您服務。



1	“我們的問題”	1
1.1	我們如何着手？	1
1.2	為什麼要研究這個問題？	2
1.3	問題的更進一步描述	3
1.3-1	控制器的輸入	4
1.3-2	控制器的輸出	5
1.4	問題的解法	6
1.5	設計的步驟	7
1.5-1	我們要解決什麼樣的問題？	7
1.5-2	規 格	8
1.5-3	發 展	10
1.5-4	實際化	13
1.5-5	測 試	14
1.5-6	操 作	14
1.6	設計週期	15
1.7	我們如何進行	16
1.8	習 題	17
2	微處理機	19
2.1	簡短的歷史	19
2.2	結 構	20
2.2-1	文紐曼結構	20
2.2-2	微處理機的結構	23

2.2-3	指令處理	25
2.2-4	堆 疊	26
2.3	指 令	27
2.3-1	指令種類	28
2.3-2	指令格式	34
2.3-3	定址模式	34
2.4	微算機的其他部份	36
2.4-1	記憶體	36
2.4-2	輸入及輸出	37
2.4-3	匯流排	38
2.5	程式規劃：基本構成概念	39
2.5-1	順 序	40
2.5-2	條 件	41
2.5-3	重 覆	42
2.6	摘 要	43
2.7	習 題	44

3 簡單的程式與MC68000 49

3.1	MC 68000 的特性	51
3.1-1	68000 的資源	52
3.1-2	程式規劃的模式	54
3.1-3	資料型態	57
3.1-4	定址模式	58
3.1-5	指令集	59
3.2	順序的例子	62
3.2-1	例題一：將 25 加到 D0 中的位元組	63
3.2-2	例題二：兩個字相加	65
3.2-3	例題三：兩字相加之另一例	66
3.2-4	例題四：三個暫存器中的字相加	68

3.2-5	例題五：將三個在記憶體中的長字相加	70
3.2-6	例題六：乘以5	71
3.2-7	例題七：對記憶體中做減法	72
3.2-8	例題八：字的除法	73
3.3	算術與數值	74
3.3-1	不帶正負號的數量	75
3.3-2	二的補數	78
3.3-3	例題九：再將25加到D0中的位元組	81
3.3-4	例題十：字的除法（不帶正負號數量）	82
3.4	條件	84
3.4-1	例題十一：重做將D0中的位元組與25相加	86
3.4-2	例題十二：字的比較	87
3.4-3	例題十三：分支程式	88
3.4-4	例題十四：字的選擇	90
3.5	摘要	91
3.6	習題	92

4 簡單的程式與iAPX 86/10 97

4.1	iAPX 86/10 的特性	98
4.1-1	iAPX 86/10 的資源	99
4.1-2	程式規劃模式	101
4.1-3	資料型態	104
4.1-4	定址模式	105
4.1-5	指令集	105
4.2	順序的例題	109
4.2-1	例題一：將25加到AL中的位元組	110
4.2-2	例題二：兩個字相加	110
4.2-3	例題三：兩字相加之另一例	112
4.2-4	例題四：將三個暫存器中的字相加	113

4.2-5	例題五：將記憶體中的三個長字相加	113
4.2-6	例題六：乘以 5	114
4.2-7	例題七：從記憶體中做減法	115
4.2-8	例題八：字的除法	115
4.3	算術與數	116
4.3-1	不帶正負號的量	117
4.3-2	二的補數	117
4.3-3	例題九：將 25 加到 AL 中的位元組	117
4.3-4	例題十：位元組除法（不帶正負號的量）	117
4.4	條件	118
4.4-1	例題十一：將 25 加到 AL 中的位元組之另一例	120
4.4-2	例題十二：字的比較	121
4.4-3	例題十三：程式分支	122
4.4-4	例題十四：字的選擇	123
4.5	摘要	124
4.6	習題	124

5 含迴路的程式 129

5.1	簡單迴路例題	131
5.1-1	例題一：延遲	131
5.1-2	例題二：另一個延遲	132
5.1-3	例題三：執行 18 次迴路	133
5.1-4	例題四：將記憶體中的十個字相加	135
5.1-5	例題五：將記憶體中的十個字相加	137
5.1-6	例題六：將記憶體中的十個字相加	141
5.1-7	例題七：將記憶體中的十個字相加 (iAPX 86/10)	144
5.1-8	例題八：搜尋最大值	145

5.2	根據我的方式來做：品質	147
5.2-1	例題九：計算係數	148
5.2-2	例題十：再度計算係數	149
5.2-3	差異處為何？	151
5.3	含有其他出口的迴路	153
5.3-1	例題十一：找尋陣列中的第一個 \$ 0 D	153
5.3-2	例題十二：再找第一個 \$ 0 D	156
5.3-3	例題十三：加總直到結束記號	158
5.3-4	例題十四：加總直到結束記號 (iAPX 86/10)	160
5.4	摘 要	163
5.5	習 題	165

6 輸入 / 輸出控制 169

6.1	I/O 控制的方法	170
6.1-1	程式控制的 I/O	170
6.1-2	中斷驅動 I/O	171
6.1-3	直接記憶體存取 I/O	173
6.1-4	使用哪一個？	173
6.2	有關中斷的一些字彙	174
6.3	MC 68000 的中斷	175
6.3-1	MC 68000 的中斷處理	175
6.3-2	MC 68000 的自向量	177
6.4	MC 68000 的例題	178
6.4-1	例題一：脈波計數器	179
6.4-2	例題二：以中斷方式做脈波計數器	182
6.4-3	例題三：輸入下一個字元	184
6.4-4	例題四：數字的輸入	185
6.5	中斷與堆疊	188