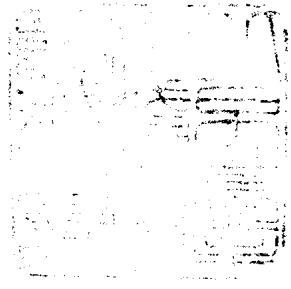
APPLE
組合語言

陸文麟　譯

協群科技出版社

APPLE
組合語言

陸文麟　譯



協群科技出版社

APPLE組合語言

編譯者：陸　文　麟
出　版：協群科技出版社
發　行：協群科技出版社
香港中環卑利街684號二樓
印刷者：廣　源　印　務　局
青山道875號工廠大廈

定價：H.K.$ 25.00

前　言

組合語言（Assembly Language）叫人害怕的唯一原因，可能就是它怪怪的，有點工業化味道的名字了；那末，爲什麼不把它當成一套"友善的語言"來看呢？因爲它正是爲您而設的啊！這本書正是爲那些需要使用並且玩賞他的Apple的普羅大衆而寫的，所以在寫作上就採取了能夠讓初學者容易了解、容易跟得上的法子。隨著這本書清晰而且漸進的方式，帶您講覽這套語言，相信您會覺得有很多人感到"組合語言不好學"這件事有點值得懷疑了。

您還可以曉您的朋友，用組合語言寫出來的程式要比用BASIC做的有時候會快上100倍!圖形、卡通、電動玩具遊戲，還有很多其它的應用，用組合語言來寫，是更爲生動的；還有，了解一點組合語言也正是開啓Apple的監督程式、DOS、與其它系統程式的大門，也爲您要充分了解Apple舖了一條坦途。

您曉得嗎？在Apple ROM的深處有一個神秘兮兮的，叫做Sweet—16的微電腦!!!!這本書還教您如何用它來減少您的組合語言程式指令數目；這可是在其它的初等教材上找不到的哦……。

目 錄

第一章 導 論…… 1

手册的目的…… 1
手册的範圍…… 1
通 論…… 1
爲何要用組合語言…… 5

第二章 符 號…… 6

概 論…… 6
位元字串…… 9
二元算術…… 14
不帶符號的整數…… 16
尼波，位元組和字…… 17
帶符號的整數…… 18
十六進位數目…… 20
底數和其毛病…… 21
ASCII字元集合…… 22
用字元串來表示指令…… 23

第三章 暫時器、指令型式和位址…… 25

概 論…… 25
累積器…… 27

X暫存器 …… 27
Y暫存器 …… 28
堆疊指標 …… 28
程式狀況字 …… 28
程式標示 …… 28
指令型式 …… 29
二位元組和三位元組指令 …… 30
6502的位址型式 …… 32
直接位址型式 …… 33
絕對位址型式 …… 33
零頁位址型式 …… 34
索引位址型式 …… 34
間接位址 …… 35
由Y間接索引 …… 35
間接的由X索引 …… 35
暗示的位址型式 …… 35
累積器位址型式 …… 36
相對位址型式 …… 36
結　語 …… 36

第四章　一些簡單指令 …… 37

新指令： …… 37
概　論 …… 37
組合語言的程式型式 …… 38
有效的標籤例子 …… 38
符號欄 …… 39
運算元欄 …… 39

註解欄 …… 40
載入群 …… 40
存入指令 …… 42
資料轉移指令 …… 43
暫存器的增加和減少 …… 44
增加和減少指令 …… 45
標示和變數 …… 45
運算元欄中的例子 …… 48

第五章　組合語言 …… 49

新指令 …… 49
概　論 …… 50
示範程式 …… 50
跳開指令 …… 52
處理機狀況暫存器 …… 54
BREAK 旗標(B) …… 54
十進位旗標 (D) …… 54
廢除中斷旗標 …… 55
情況碼旗標 …… 55
轉移指令 …… 57
迴　路 …… 58
比　較 …… 60
IF/THEN 陳述的模擬 …… 62
FOR/NEXT迴路 …… 62
布林值的測試 …… 66

第六章　算術運算 …… 69

新指令 …… 69

概　論…… 69
不帶符號整數（二元數）算術…… 69
不帶符號加法的規則…… 72
減法…… 72
不帶符號減法的規則…… 73
帶符號算術…… 73
帶符號算術的規則…… 75
帶符號的比較…… 75
二元碼十進位算術…… 76
不帶符號的 BCD 算術…… 77
十進位算術範例…… 78
不帶符號算術的規則…… 78
帶符號的 BCD 算術…… 79
摘　要…… 79
8 位元算術的規則…… 80

第七章　副程式和堆疊處理…… 81

新指令…… 81
概　論…… 81
變數問題…… 83
傳送參數…… 93

第八章　陣列第零頁索引和間接位址法…… 95

新指令…… 95
概　論…… 95
第零頁位址法…… 95
組合語言中的陣列…… 97
在編譯時給定陣列初值…… 102

用索引位址法來處理陣列元素 …… 104
間接位址法 …… 106
間接索引位址法 …… 110
索引間接位址法 …… 111

第九章　邏輯罩幕和位元運算 …… 113

新指令 …… 113
概　論 …… 113
補數函數 …… 114
AND 函數 …… 114
OR 函數 …… 115
XOR 函數 …… 116
位元字串運算 …… 117
邏輯運算的指令 …… 118
AND 指令 …… 118
ORA 指令 …… 119
XOR / EOR指令 …… 120
取累積器的補數 …… 120
罩幕運算 …… 121
罩　除 …… 121
罩　進 …… 126
移動和旋轉指令 …… 127
算術往左移（ASL ）指令 …… 127
邏輯往右（LSR ）指令 …… 129
往左旋轉（ ROL）指令 …… 130
往右旋轉（ ROR）指令 …… 131
移動和旋轉記憶體位置 …… 131

用 ASL 來作乘法……132
用移動來聚集資料……134
用旋轉和移動來聚集資料……135

第十章　多重精確度運算……137

概　論……137
多重精確度的邏輯運算……137
多重精確度的移動和旋轉……139
多重精確度的邏輯往右移動系列……140
多重精確度的往左旋轉……141
多重精確度的往右旋轉……142
多重精確度的不帶符號算術……143
N位元組不帶符號的加法規則……144
多重精確度的無號減法……145
二位元組的減法範例……145
多重精確度減法的規則……145
多重精確度的有號算術……146
多重精確度的十進位算術……146
多重精確度的增加……146
多重精確度的減指令……147
多重精確度的不帶符號比較……148
測試——16位元值是否爲0……148
測試——16位元值是否爲負數……149
相等和不等的測試……149
帶符號的比較……152

第十一章　基本的輸出輸入……155

概　論……155

字元輸出 …… 155
標準輸出和週邊裝置 …… 163
字元輸入 …… 165
一行字元的輸入 …… 168

第十二章　數值的I/O …… 171

概　論 …… 171
16 進位的輸出 …… 171
十進位值的位元組輸出 …… 173
16 位元不帶符號整數的輸出 …… 175
帶符號16位元整數的輸出 …… 176
輸出整數的一種簡單方法 …… 177
數值輸入 …… 178
16 進位和BCD …… 178
不帶符號十進位的輸入 …… 181
帶符號十進位的輸入 …… 187

第十三章　乘法和除法 …… 191

概　論 …… 191
乘　法 …… 191
除法演算法 …… 191

第十四章　處理字串的運算 …… 203

字串的處理 …… 203
宣告字串常數 …… 207
字串設定 …… 207
字串函數 …… 209

字串串連……211
次字串運算……213
字串的比較……215
字元陣列的處理……220

第十五章　APPLE Ⅱ 特殊的輸入方式……225

APPLE輸出入結構……225

第十六章　SWEET-16簡介……235

SWEET-16……235
SWEET-16硬體的要求……246

第十七章　程式的偵錯與除錯……247

GO-指令(G)……248
暫存器與記憶體的起始值……249
修正指令碼（補正）……253
程式偵錯例子……257

附　錄……261

第一章
導 論

手册的目的（*PURPOSE OF MANUAL*）

這本手册提供了 APPLE II 機器可用的6502組合語言的指令。所包含的內容適用於初級，中級和高級程式設計師。

手册的範圍（*SCOPE OF MANUAL*）

書中包括了基本符號和常用術語的解釋。也包括計算機概念的介紹，簡單的組合語言例子及與 APPLE II 有關的6502組合語言指令的介紹。

通 論（*GENERAL*）

爲什麼要另外寫一本專講6502 組合語言的書呢？原因是第一，目前這方面的書只有二本。第二，其中沒有一本是專門針對 APPLE II 計算機而討論的。雖然你可從這些書中學到組合語言的理論，可是對擁有 APPLE II 的使用者而言

，書中的範例卻一點用也沒有。

這本書是我做爲 6502 組合語言指導時所得經驗的累積。書中的內容對初學者而言並不難。雖然不能保證看完此書能精通到何種程度，但可以確定的是你一定可以成爲具有中等程度的 6502 組合語言的程式員，“精通”的階段仍是需要經過幾年的經驗才可達到的。

假如你曾用過 6502 ，則頭幾章所講的你可能已經知道了。但不要跳過任何一段！某個重要的細節若不詳細了解，則會影響到整本書其他部分的了解程度。因此要看過書中所有的資料且在繼續下去之前要確信你已了解了複習的部分。假如你是初學者，則一定要了解每一節之後才能繼續看下去!

有很多關於計算機理論和微計算機的書，因此我把關於這些的討論儘量減少。假如你對 6502 組合語言很有興趣下列的書我推薦你應該去買來看：

HOW TO PROGRAM MICROCOMPUTERS
by William Barden Jr.

PROGRAMMING THE 6502
by Rodney Zaks

PROGRAMMING A MICROCOMPUTER
by Caxton C. Foster

6502 ASSEMBLY LANGUAGE PROGRAMMING
by Lance Leventhal

6502 SOFTWARE GOURMET GUIDE & COOKBOOK
by Robert Findley

雖然前面幾本書都很好，可是都不是與 APPLE II 直接相關的。假如你想學好組合語言則你該看過前面所提的幾本書，和這本手册。

在討論組合語言之前，先讓你熟悉一些以後經常會常用到的名詞：

RAM （隨機處理記憶體）：使用者可以使用的記憶体。程式和資料都存在RAM中。

ROM （僅讀記憶體）：用來放APPLE的監督程式（monitor）和BASIC的地方。使用者不可將程式或資料存在此處。

MONITOR（監督程式：僅讀記憶體中的一組副程式，可以使你讀入鍵盤輸入的資料，在銀幕上顯示出字元等。

BASIC：代表整數BASIC。

K：當看到K時，就把它代換成" × 1024 "（即乘以1024），通常用來表示記憶體的大小。（如48K）。

記憶體（memory）：爲所有RAM和ROM的組合。

帶符號數字：任何合法的正整數或負整數（在目前運算下合法的數字）。

不帶符號數字：任何合法的正整數。不允許使用負整數。

位元組：記憶體的一種單位。一個位元組可代表256種不同的數值（如0-255之間的整數）。

字（WORD）：由二個位元組連成的。用一個"字"，可代表65,536種不同的數值（如0～65535之間的整數，或－32768～32767之間的帶符號數字）。

文法（syntax）：程式語言中用來規定句子結構的法則

位址（address）：爲二個位元組所組成的，用來指到64K個可用記憶位置中的某一個。一個位址也是一

個字，但一個字不一定是一個位址。

頁（ PAGE ）：APPLE Ⅱ 計算機中65536個位元組被分成256個片段，每個片段由256個位元組組成。每個片段稱爲一頁，編號從0到255 。

第0頁：記憶體中最前面的256個位元組稱爲第0頁。當然也有第一頁、第二頁，可是在機器中最常用的是這一個片段，因此特稱爲第0頁”。

擴充接點（ solt）：APPLE Ⅱ 計算機中具有8個與週邊設備連接的擴充接點。

I/O：輸入/輸出。

LISA ：爲Lazer System Interactive Symbolic Assembler（雷射系統接觸式組合編譯器）的簡稱。

週邊裝置（ peripheral ）：與計算機相連接的外部輸出入裝置。

假設你已熟悉 Apple 的 BASIC 語言。本書中只在一些例子中會用到 BASIC，但如你已熟悉 BASIC 那表示你已具有基本的程式設計技巧了。你應先了解一些程式技巧之後再來學組合語言。組合語言牽涉較廣，如果你一邊學最基本的程式技巧一邊學組合語言，則很容易會使你自己處於混亂的狀況下。

學習任何程式語言，尤其是組合語言，都要有實際操作的經驗。書中的例子都用 LISA（爲 APPLE Ⅱ 的6502組合編譯器）。 LISA 對初學者最適合，因爲它是接觸式的，也就是當每一行程式進入機器後，系統會立刻抓出文法上的錯誤。這與 APPLE Ⅱ 中的整數 BASIC 很像。其餘 APPLE Ⅱ Ⅱ 可用的組合編譯器都沒這樣的功能。

爲何要用組合語言
(*WHY USE ASSEMBLY LANGUAGE?*)

當速度是程式中首要的要求時，或是當你需要控制某種週邊裝置時或是利用高階語言無法完成你的應用問題時，就要使用組合語言。

不要把組合語言用在商業或科學用途上，因爲在這方面的工作中， Pascal、FORTRAN 或 Applesoft 會更適用。浮點的運算雖然不是不可爲，但可能太難，不是初學者或中級程式員能夠處理的。

組合語言所提供的另一個好處是可與現有的 BASIC 或 Applesoft 或 Pascal 的程式互相連接。你可以用組合語言寫時間要求較嚴格的部分，而其他部分則以 BASIC 來寫。

一旦你對 組合語言熟悉了之後，你會發現它的編寫和除錯與 BASIC 程式的編寫和除錯一樣地簡單！