

8080 8085

軟體設計

上 冊

陳順成 編譯



8080/8085 軟體設計

上 冊

陳順成 編譯

協群科技出版社

8080/8085軟體設計 上 冊

編 譯 者： 陳 順 成

出 版： 協 群 科 技 出 版 社

發 行： 協 群 科 技 出 版 社

印 刷 者： 廣 源 印 務 局
香港中環卑利街684號二樓
青山道875號工廠大廈

定價： H.K.\$ 45.00

序 言

當 Intel 公司首次宣佈四位元的 4004 微處理機晶片、把微處理機與微電腦的觀念公佈於世時，當時這似乎並不是科技的重大革命，不過由於積體電路技術的進步，及更多的公司投入微處理機的生產，微電腦對整個工業界已造成了大革命，現在這種由淨化過的“沙”所做出的化學元件，真的改變了我們的生活方式。

在超級市場電子收銀機不只可以取代以前機械式收銀機的工作，也可以做盤存的工作，有些縫紉機不再需要複雜的傳動裝置，而改由微電腦控制針的位置，即可刺繡一個特殊的花樣。在最近的未來，您就可以教縫紉機刺繡最新的花樣。

我們已不再使用計算尺作複雜的計算，計算器具有更好的計算功能。手錶不再需要上發條，順時鐘和逆時鐘方向都將成為歷史名詞。今天您可以使用電子支票簿，管理在銀行的存款，也不再需要每個月計算支票簿了。在不久的將來，微波爐會比您更懂得烹調。

使用微電腦解決問題需要硬體和軟體，硬體包括微處理機積體電路、記憶、LED、和一些介面電路，使微電腦可以控制週邊裝置（燈、馬達、閥、線圈、或儀表等）。軟體包括一些微電腦執行的指令序列，讓微電腦可以處理資料或控制週邊裝置，不幸的是，軟體的設計和製造之定義並不像硬體那麼明確。

目前有些微電腦用者以 BASIC 語言寫程式，這樣需要一個 BASIC 解釋元，一個由組合語言寫成放在記憶中，它約需佔用 5000 到 8000 個記憶位置，這個組合語言程式可以讓您把您的

BASIC 程式輸入微電腦，並且真正地執行它。和世界上的每一件事情一樣，以 BASIC 語言寫程式有利也有弊，基於使用 BASIC 語言的缺點，因此在微電腦程式上，經常使用組合語言，本書即專門討論這種語言。

組合語言毫無疑問地對教者和學者都是比較困難的，您不可能以組合語言在 10 或 15 分鐘內寫出可以把 1 到 1000 的所有數開立方根的程式，使用 BASIC 語言就可以。但是有很多工作，無法以 BASIC 語言完成，而必須使用組合語言，特別是程序控制、週邊控制、高速計算、和即時資料收集等方面。事實上絕大部份以微電腦設計的消費性產品都是以組合語言寫成，例如：電視遊戲、微波爐、縫紉機、電子收銀機、汽油泵、和血壓計等，因此組合語言是非常強而有用的語言，值得學習。

本書第一章中，我們不討論任何組合語言程式，而介紹 8080 和 8085 微處理機積體電路的特性：包括這兩個積體電路內所有暫存器的介紹，這些暫存器是非常重要的，在組合語言程式中要反覆地使用它們。在接下去的三章中，要討論 8080 和 8085 所能執行的組合語言指令，在前面這幾章沒有長的程式，因為您尚未熟悉指令，所以無法了解長而複雜的程式。充分了解這些指令後，您就可以開始以組合語言寫微電腦程式，使其執行有用的工作。在下面幾章中，將分別討論 8080 組合語言在數學運算和輸入 / 出裝置（週邊裝置）控制上的運用。

本書有許多您會喜歡的特點。本書中，我們並未專門闢出一章討論包括二進、八進、十六進、與十進制數字系統。但是在幾章中，我們改用紙和筆來說明如何作數字系統之運算，我們較有興趣的是如何以微電腦執行這些運算。如果您需要更多有關於基本數學運算的資料，可以在其他計算機程式書上找到。您也將發現本書的程式將可以在任何 8080

微電腦系統上執行，我們並沒有特別舉出製造廠商出品的硬體和軟體的例子。不過這些軟體範例一樣的可以在MITS、Intel、NS、Digital Group、Control Logic、IMS或E&L儀器等以8080為主的微電腦上執行。當然各家系統的週邊裝置都不一樣，裝置地址也不一樣。

第七章（輸入／出）有軟體所用之裝置的電路圖，不過除非您能了解週邊裝置和微電腦介面，否則控制週邊裝置的軟體程式毫無意義，而且如果您想要在自己的微電腦上執行某個程式，您或許將需要週邊裝置的接線圖以複製出硬體。

在前三章中，我們討論各種指令的八進制和十六進制操作碼。我們比較喜歡八進制碼，因為8080的組合語言操作碼，以八進制表示比較容易記住；檢查八進制的操作碼，也可以很容易地看出指令所影響到的暫存器或記憶位置，如果使用十六進制表示，則在這方面顯得較為困難。另一方面，很多程式師喜歡十六進制的數系統，因為兩位的操作碼要比三位的操作碼容易記住。因此我們的結論是兩種數系統都使用。我們建議8080或8085的初學者使用八進制碼，因為它比較容易記住，以後用十六進制則比較方便。如果您不知道用那個數系統好，兩者都試試，找出您最適合的數系統。在本書中十六進制的數通常都用括號括起來，例如：（5F）。

本書的另一個目標是提供程式動作之詳細解說，而不是要您自己去想程式如何工作。如果用後面這種方式，您會學得很少。我們的方式是針對某項特殊任務，以最簡單的指令序列開始，當然這些程式會有某些限制，不過我們可以慢慢增加指令，讓微電腦漸次完成其所負的任務。因此我們是由學習以前的例子，設計最佳方式來解決目前之問題。

軟體的好處之一是它印出與分發很簡單，有鑑於此，許多專業

性或業餘性的電腦雜誌提供程式的原始列表和標的碼紙帶或錄音帶。軟體程式列表在Electronics Design、Computer Design、及Electronics 上常有。業餘性雜誌Interface Age (Mcpheters, Wolfe & Jones, Cerritos, CA 90701), SCCS Interface (SCCS, Santa Monica, CA 90405), Kilobaud and Byte (both of Peterborough, NH 03458) 和Dr. Dobbs Journal of Computer Calisthenics and Orthodontia (Menlo Park, CA 94025) 上也有很多程式列表。其他有一些軟體圖書館，包括Intel User's Library, 3065 Bowers Ave., Santa Clara, CA 95051, 及Microcomputer Software Depository, 2361 E. Foothill Blvd, Pasadena, CA 91107. Depository 的新增資料每個月都會列在Interface Age 上。

Tychon 公司專門在訓練科學家、工程師、和電子 / 計算機愛好者，課程範圍是數位電子、微電腦硬體和軟體。我們在American Laboratory (International Scientific Communications, Inc., 808 Kings Highway, Fairfield, CT 06430), 泰國刊物Semiconductor Electronics Journal (Science, Engineering and Education CO., Ltd., Bangkok, Thailand), 德國刊物, Elektroniker (Burgdorf, Switzerland) 上都有每月專欄。在Electronic News (IPC Business Press Pty.-Ltd., Sydney, Australia), Ham Radio (Greenville, NH), Computer Design (Concord, MA), 及Radio Electronics (New York City, NY) 上也有專欄。我們的書正在被翻譯成義大利文、德文、法文、西班牙文、和泰文。

如果您對基本數位電子學有興趣，Blacksburg Continuing Education Series 有兩本書，書名是Logic & Memory Ex-

periments Using TTL Integrated Circuits，書中介紹這個主題並給一些實驗，另一本書是 Interfacing and Scientific Data Communications Experiments，專門為工程師、科學家、和業餘嗜好者而寫的，介紹如何使用 UART 做非同步通信，也有使用非同步通信技巧與電腦通信介面的實習。這個系統有三本書是有關於 8080 微處理機 / 微電腦：The 8080A Bugbook：Microcomputer Interfacing and Programming，和 Introductory Experiments in Digital Electronics and 8080A Microcomputer Programming and Interfacing 上下兩冊，這幾本自修的書籍中說明基本的介面硬體和軟體、如何使用、如何以軟體取代硬體、及其他許多有趣的主題。我們最新的書是 Microcomputer-Analog Converter Software and Hardware Interfacing，此書說明處理微電腦的數位對類比和類比對數位轉換器之原理。

目 錄

第一章	8080/8085微處理機介紹	1
•	8080 微處理機	2
•	機器與組合語言	6
•	8085 微處理機	9
•	本書慣用之表達方式	10
•	程式之格式	12
•	8080 與 8085 相似之處	13
•	8080/8085 指令整理表	13
第二章	8080/8085基本指令	121
•	資料移動指令	122
•	使用讀 / 寫記憶儲存資料	125
•	立即資料移動指令	127
•	簡單的暫存器對指令	131
•	輸入和輸出指令	135
•	八位元邏輯與數學指令	139
•	邏輯指令	142
•	數學指令	152
•	分支、控制的轉移、及作決定指令	169
•	結論	185

第三章 副常式與基本指令的使用	187
• 叫出副常式	189
• 時間延遲副常式	195
• 條件叫出與返回指令	204
• 使用基本指令	210
• 入 / 出裝置同步	210
• TTY 入 / 出與字元處置	218
• TTY 及終端機程式	229
• 電子鎖	235
• 結論	240
第四章 8080/8085的高等指令	243
• 暫存器對的運算	243
• 堆疊指標暫存器	249
• DAD 指令	250
• 直接取和存的指令	256
• 使用堆疊存放資料、地址、與狀態訊息	262
• 重始指令——單位組叫出指令	270
• 使用H暫存器對	274
• 其他的A暫存器（累積器）指令	286
• 進位指令	288
• 結論	290
第五章 數學常式	293
• 整數加法	293

• 整數減法	303
• 整數乘法	308
• 整數除法	320
• BCD 數學	334
• 四位數 BCD 運算	340
• 浮點數學運算	347
• 特殊函數	354
第六章 數基底轉換	357
• 三位數、ASCII 基底、八進制對二進制轉換	357
• 八位元、二進制對 ASCII 基底的八進制轉換	363
• 二位數、ASCII 基底、十六進制對二進制轉換	369
• 八位元、二進制對 ASCII 基底的十六進制轉換	375
• 三位數、以 ASCII 為基底、十進制對二進制轉換	379
• 八位元、二進制對 ASCII 基底的十進制轉換	389
• 十六位元、二進制對 ASCII 基底的十進制轉換	394
• 是否需要轉換的問題	399
• 在計數器程式和副常式中使用 DAA 指令	405
• 去掉前面的零	408
• 結論	410
第七章 微電腦輸入/出	413
• 入 / 出資料移轉——匯流控制	417
• 8080 與簡單的入 / 出裝置	418
• 8080 與鍵盤	423
• 使用硬體編碼器的鍵盤所需之軟體與硬體	423

• 軟體驅動、多工（掃描）式鍵盤.....	435
• ASCII 基底式鍵盤與 8080 之介面.....	445
• 8080 和 LED 顯示器.....	447
• 記憶對映入 / 出.....	462
• 具有硬體編碼器的記憶對映入 / 出鍵盤.....	465
• 記憶對映入 / 出、多工（掃描）式鍵盤.....	467
• 記憶對映入 / 出 LED 顯示器.....	471
• 記憶對映入 / 出 10 位數多工式顯示器.....	475
• 結論.....	476

名詞對照表	478
-------------	-----

程 式 範 例

第一章

1 - 1	典型的 8080 程式.....	6
1 - 2	一些 8080 指令的代號及操作碼.....	8
1 - 3	程式列表之格式.....	12

第二章

2 - 1	把相同的值放入暫存器 B、C、D 和 E 中.....	128
2 - 2	把一個資料位組存入四個暫存器的兩種方法.....	128
2 - 3	把地址為 030 123 (1853) 的記憶中的資料移入 D 暫存器.....	130
2 - 4	把立即資料位組存入記憶中.....	131
2 - 5	LXI H 和功能相當的 MVI 指令.....	132
2 - 6	使用 LXI B、LXI D、LXI H 指令.....	133
2 - 7	送八位元的值到地址為 015 (0D) 的裝置.....	136
2 - 8	從地址為 103 (43) 的裝置輸入八位元的資料.....	136
2 - 9	遮掉 ASCII 字元中的四個 MSB	148
2-10	使用 AND C 指令遮掉四個 MSB	148
2-11	把兩個 ASCII 字元輸入、遮蓋、旋轉及合成.....	150
2-12	使用 ADD B 指令 把 B 暫存器內的資料加入 A 中.....	153
2-13	把 A 和 B 暫存器內的資料相加的程式.....	154
2-14	把 A 和 B 內的資料相加並產生進位.....	154

2-15	本程式使兩數相加且結果設置進位位元為 1	155
2-16	兩個 16 位元數的加法	156
2-17	使用 ADC 指令作 16 位元加法	157
2-18	從暫存器 A 中減去暫存器 E 中的資料	158
2-19	B 暫存器減以 E 暫存器	158
2-20	減法中產生借位	159
2-21	小數減以大數	159
2-22	兩個 16 位元數的減法	159
2-23	D 暫存器對減以 B 暫存器對	160
2-24	D 暫存器對減以 B 暫存器對，有產生借位	160
2-25	D 暫存器對減以 B 暫存器對	161
2-26	以 INR B 指令增加 B 暫存器內的資料	163
2-27	以 DCR E 指令減去 E 暫存器中的資料	163
2-28	使用 INX H 指令	164
2-29	把記憶內的資料移入暫存器 D 和 E	165
2-30	例 2-29 中程式的改進型	165
2-31	說明 DCX H 指令	166
2-32	使用 HLT 指令	168
2-33	在程式中使用 NOP 指令保留空間	169
2-34	跳越指令的格式	170
2-35	跳回程式的開始點	171
2-36	使用問號來結束輸入程式	173
2-37	去掉除 0 - 9 及 ? 以外的全部 ASCII 字元	174
2-38	先測試 ASCII 數字字元	176
2-39	使用旋轉指令及進位旗號測試選擇的位元	177
2-40	使用 ANI 指令和零旗號來測試 A 暫存器中選擇的位元 ...	178

2-41	以旋轉指令測試一個字中的多個位元	180
2-42	依次測試資料位元 D_4 、 D_3 及 D_4	181
2-43	在一個字內使用 ANI 指令測試三個資料位元	182
2-44	等到某個資料位元變成零	183
2-45	位元 D_2 的設置運算	184
2-46	位元清除或位元重定指令	185

第三章

3-1	以 LXI SP 指令把值放入堆疊指標中	192
3-2	200 毫秒的時間延遲副常式	198
3-3	30 秒的時間延遲副常式	200
3-4	簡化的 30 秒時間延遲副常式	201
3-5	使用暫存器對減一指令的 0.2 秒時間延遲副常式	202
3-6	叫出 HAF-MIN 副常式的一小時時間延遲程式	203
3-7	印出 ASCII 字元的程式 (記憶中不存跳行)	205
3-8	二進制對 ASCII 字元轉換 (十六進制數字) 副常式	208
3-9	簡單的 TTY 輸出副常式	211
3-10	在 TTY 上印出一個 B 字元	213
3-11	印出字元的彈性方式	214
3-12	把鍵盤字元的回音印在印字機上	215
3-13	在入/出軟體中使用 ANI 指令, 富有彈性	217
3-14	輸入 ASCII 字元並存入記憶中	218
3-15	能得到資料回音的輸入及保存程式	220
3-16	用問號表示資料輸入完畢	221
3-17	若輸入回轉, 則加印跳行	223
3-18	把存在記憶中的 ASCII 字元印出	224

3-19	印出 ASCII 資料(以 000 表示資料的結束)	225
3-20	一般用途的 TTY 入/出副程式	227
3-21	如何印出 CR、LF、和 BELL	229
3-22	TTY 或 CRT 測試程式	230
3-23	打紙帶機測試程式	232
3-24	讀紙帶機測試程式	233
3-25	電子鎖程式	235
3-26	改進的電子鎖程式	238

第四章

4 - 1	搬動一塊資料	245
4 - 2	改進的一塊資料移動程式	246
4 - 3	由高往低搬移重疊的陣列	248
4 - 4	兩個 16 位元的數相加	251
4 - 5	執行 20 次的 DAD H 指令	253
4 - 6	找出堆疊指標中的地址	254
4 - 7	把 SP 作加及減的運算	255
4 - 8	把 H 暫存器對中的資料存到 R / W 記憶中	257
4 - 9	SHLD 指令使用方法	257
4-10	執行 DAD SP 指令之前先使用 SHLD 指令	258
4-11	以 LHLD 指令把資料放入暫存器對 H 中	259
4-12	測驗您對 LHLD 指令了解程度	259
4-13	找出 SP 中的地址而不破壞暫存器對 H 中的資料	260
4-14	使用 LDA 指令	261
4-15	使用 STA 指令	261
4-16	判斷記憶中是否存有 215(8D)	262

4-17	使用堆疊來存暫存器中的資料之正確方法	265
4-18	把資料壓入及彈出堆疊	267
4-19	何時壓入及彈出	269
4-20	不應該這樣子叫出副常式	270
4-21	使用重始指令叫出副常式	272
4-22	使用叫出指令取代重始指令	273
4-23	使用重始指令叫出長的副常式	273
4-24	使用 XCHG 指令	275
4-25	不用 XCHG 指令，只用 LXI 指令	275
4-26	與 XCHG 指令功能相當的方法	275
4-27	把暫存器 D 存入 R / W 記憶中	276
4-28	把暫存器對 D 存入 R / W 記憶中的改進方法	276
4-29	使用 LHLD 指令，把記憶中的資料放入暫存器對 D 中	277
4-30	使用 MOV 指令把記憶中的資料放入暫存器對 D 中	277
4-31	把暫存器對 D 和 H 存入記憶中	278
4-32	把暫存器對 B 中的資料存入記憶中	278
4-33	B 暫存器對與 D 或 H 暫存器對互換	279
4-34	資料陣列移動程式	280
4-35	使用 LHLD 指令取得地址和數目	281
4-36	使用 PCHL 指令	282
4-37	使用 XTHL 指令	283
4-38	使用 SPHL 指令	284
4-39	存於 ROM 中的程式能夠改變 SP	285
4-40	簡單的 CMA 指令	286
4-41	另一個使用 CMA 指令的程式	287
4-42	產生 2 補數	288