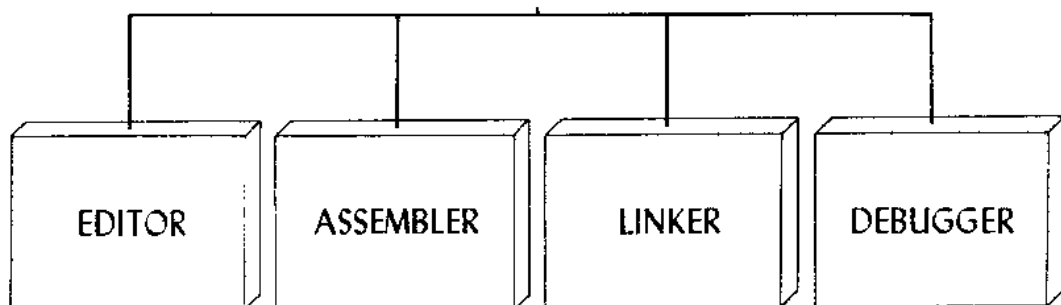
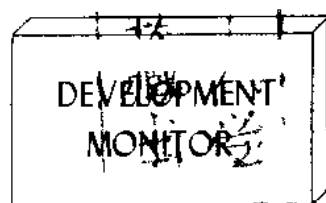


PA800微型電腦 組合語言實習手冊



全亞電子工業股份有限公司編寫
全華科技圖書公司印行



全華圖書 版權所有 翻印必究
局版台業字第0223號 法律顧問：陳培豪律師

PA800微型電腦 組合語言實習手冊

全亞電子工業股份有限公司編寫
總公司：台北市基隆路二段43號
電話：7038443・7088206・7083591
台中辦事處：台中市稱許五街11號
電話：(042)554688
高雄辦事處：高雄市七賢一路225號11樓5室
電話：(07)2821476

出版者 全華科技圖書股份有限公司
北市龍江路76巷20-2號
電話：581-1300・541-5342
581-1362・581-1347
郵政帳號：100836
發行者 陳 本 源
印刷者 華一彩色印刷廠
定價 新臺幣 180 元
再版 中華民國72年8月

序

不斷的研究開發，推廣微電腦的應用，是全亞公司一貫的宗旨，全亞電子公司成立至今有六年多，一直爲了提高國內微電腦技術水準而努力，多年來由於全體同仁的辛勤耕耘及國內外廣大客戶的支持，使得公司得以迅速成長。從“微電腦”這個名詞剛剛在國內出現時，全亞即以國內第一家微電腦專業公司的形態爲國內提供專業化之服務，其後相繼有國內第一部微電腦學習機EDU-80、工業順序控制專用電腦PASC-DT、PASC-1順序控制語言、PA800工業控制電腦、PA800B教學專用電腦、PA1000磁碟電腦系統，多人共用電腦系統的推出，每樣新產品的推出都代表著全亞公司的另一個升級，及國內電腦工業水準又邁向另一個里程，以國人的智慧、國人的技術，從微電腦學習機到大量資料處理的多人共用電腦系統，無論電腦教育訓練、工業控制、產品開發、行政事務處理、商業資料處理，全亞公司提供國人最佳的品質，最佳的服務及最合理之價格，希望能爲我國的工業升級略盡棉薄之力。

全亞電子公司所推出的PA800、PA800B微型電腦，是爲工業控制產品發展及教學用途而設計的，可執行Z-80組合語言、BASIC-C控制用及BASIC-B商業用之便捷語言，提高電腦的適用性，硬體線路採用STD-Z80 BUS模板化的組合。以組合語言代替機械碼的輸入，不僅節省機械碼翻譯的時間，減少翻譯過程中所產生的錯誤，並可自動連結許多不同程式，組成一功能強大的操作程式，真正達到微電腦應用的目的。以STD-Z80 BUS模板化的設計，使學生可以直接做各種微電腦控制應用，毋須再做微電腦電路板，隨著不同模板的功能，使PA800系列微型電腦，不僅具有數位TTL輸出入控制、類比信號輸出入控制，尚有多種工業控制功能。全亞公司期盼PA800微型電腦的推出，能夠迅速有效的推廣國內微電腦的應用，齊爲國內工業升級而努力。

PA800、PA800B 微型電腦，具有執行 Z-80 組合語言的功能，所提供的公程式計有編輯程式 (EDITOR)、編輯程式 (ASSEMBLER)、連接程式 (LINKER)，偵錯指令可供使用、執行程式輸入、執行、偵錯之用，是最適合初學者或設計工程師用的電腦，全亞電子公司為方便使用者能很快學會組合語言及機器操作技巧，以發揮 PA800、PA800B 微型電腦簡單易學之特性，特別投入大量人力編成這本中文的組合語言實習手冊，希望能提供初學者為學習指南，或學校學生的實習教材。

本教材共分二大部份，第一部份為基礎篇，第二部份為應用篇，基礎篇區分為十七個實習，教材內容之安排偏重於組合語言指令之熟悉與程式規劃 (PROGRAMMING) 之技巧練習，應用篇則偏重於實際之應用，教導學生如何運用微電腦來做各種控制與應用，達到真正學以致用的目的。

本教材內容豐富適合高工、大專院校之學生一學期或一學年每星期三～六小時之微電腦組合語言實習之用，如安排一學期課程，則以基礎篇為實習教材，如開一學年的課程，則可增加應用篇為實習教材，學生只要依序練習，配合 Z-80 組合語言課堂講授，則課堂上的理論不但馬上可以得到驗證，同時電路應用實習也可使學生建立硬體觀念，是一本軟體、硬體兼顧的組合語言實習教材。

本書之編撰，力求完美，但疏漏之處在所難免，盼望讀者及先進們閱後不吝指教，不勝感激。最後，要特別感謝陳登旭先生對本書之編撰出力最多。

全亞電子工業股份有限公司 謹識

感謝您

感謝您選購全華圖書！

希望本書能滿足您求知的慾望！

圖書之可貴 在其量也在其質

量指圖書內容充實、質指資料新穎够水準，我們就是本著這個原則，竭心盡力地為國家科學中文化努力，貢獻給您這一本全是精華的全華圖書。

目 錄

實習準備	1
一、基礎篇	
實習一 PA800 - DM開機與基本試驗	9
實習二 基本DM操作指令實習	19
實習三 編輯程式使用實習	24
實習四 程式的執行	33
實習五 磁帶資料存取程式	48
實習六 基本算術與邏輯操作指令之運用	57
實習七 加減法練習	66
實習八 跳線指令和程式迴路	76
實習九 堆疊記憶與副程式	83
實習十 旋轉、移位指令和乘程式	90
實習十一 程式執行時間和記憶位址佔用	97
實習十二 二進制除法程式	101
實習十三 二進制對十進制轉換程式	106
實習十四 十進制對二進制轉換程式	110
實習十五 開平方程式	117
實習十六 程式之中斷控制	124
實習十七 Z - 80 CTC之使用	130
二、應用篇	
實習十八 外接SEVEN SEGMENT之顯示練習	137
實習十九 紅綠燈控制練習	142
實習二十 MICROCOMPUTER ORGAN之應用	149
實習二十一 TV GAME 程式之撰寫	155
實習二十二 數位與類比資料之轉換	166
實習二十三 鍵盤與螢幕使用	177
實習二十四 中國文字顯示	182
實習二十五 數位IC功能測試	190
實習二十六 步進馬達控制	195
實習二十七 數字時鐘	208
實習二十八 PAB - PROG BOARD之使用	216

附錄一	Z - 80 CPU指令群.....	224
附錄二	PA800 UTILITY SUBROUTINE.	235
附錄三	EDU85 EDU86 及 EXTENDER BOARD 用法介紹.....	240
附錄四	PA800 DM ERROR MESSAGE.....	247
附錄五	ASCII CODE - CHARACTER TABLE.	251
附錄六	PROG 綫路圖.....	252

實習準備

程式 (Program) 是由一連串有秩序有計劃安排之指令組合而成。CPU 根據程式執行一連串之邏輯動作，以產生吾人所期待之結果。

程式在 CPU 執行之前必然以 0 及 1 的型態存在於記憶組 (Memory) 中，吾人稱此型態之程式為機器語言程式 (Machine language Program)，為了書寫方便通常以十六進制數字表示之，例如 Z 80 CPU 之 8 bit 指令 10101111 B (B 代表 BINARY) 以十六進制表示則為 0AFH (H 代表 HEXADECIMAL)，第一個字為阿拉伯數字表示下面為數字資料。

以機器語言書寫程式不容易看出其實質上之意義，所以各微處理器製造廠家都將其 CPU 指令，按特性加以分類，以具有意義的文字符號來代替每一個機器語言指令，例如 Z 80 CPU 指令 70 若以文字 LD A, L (Load data into Reg. A from Reg. L) 來表示則一目了然，這種文字寫法可用另一編輯程式 (Assembler) 將其組合成 CPU 可接受之機器語言，稱之為組合語言程式 (Assembly language)。

以組合語言書寫程式固然方便，但是要輸入記憶組供 CPU 執行還是必須先將其編譯成機器語言。在 PA 800 微型電腦系統中，吾人可利用其內存編輯程式 (resident Assembler) 來將組合語言程式編譯成機器語言，再供 CPU 執行之。

不論是多繁複的程式，都是由許多小程序式組織而成，或者說是由主程式配合許多副程式組合而成。指令的數目雖然有限，但是其組合却幾近於無限。所以軟體的設計是千變萬化的。

同一問題有無數種解法。由於要求的條件與限制條件的不同，程式設計者應配合擁有的條件做出較適切的設計，圖 1-1 表示一般微電腦程式之設計過程。從問題分析開始至完成程式設計也因為問題與要求的不同而繁簡不一，但設計者應多充實相關知識，磨練程式設計能力則是完成較佳設計必要的條件。

一、問題分析

簡單的問題只要簡單的將題目之輸入與輸出及要求功能下定義清楚即可着手程式設計，或者只要將現有程式稍微修改，或者套用與組合，但是較複雜的程式譬如監督程式、編譯程式、或控制與特殊應用的程式則需要週詳的分析後方能着手程式設計，一些必要分析的項目如下：

- (1) 問題的要求與性質。
- (2) 已知的條件。
- (3) 資料來源的型式與轉換。
- (4) 資料輸出的型式與轉換。
- (5) 資料的種類與精確度。
- (6) 執行時間的要求。
- (7) CPU 的性能與指令。
- (8) 記憶組容量的大小。
- (9) 解答的可能性。
- (10) 解答的方案。
- (11) 價值分析。
- (12) 結果的處理等。

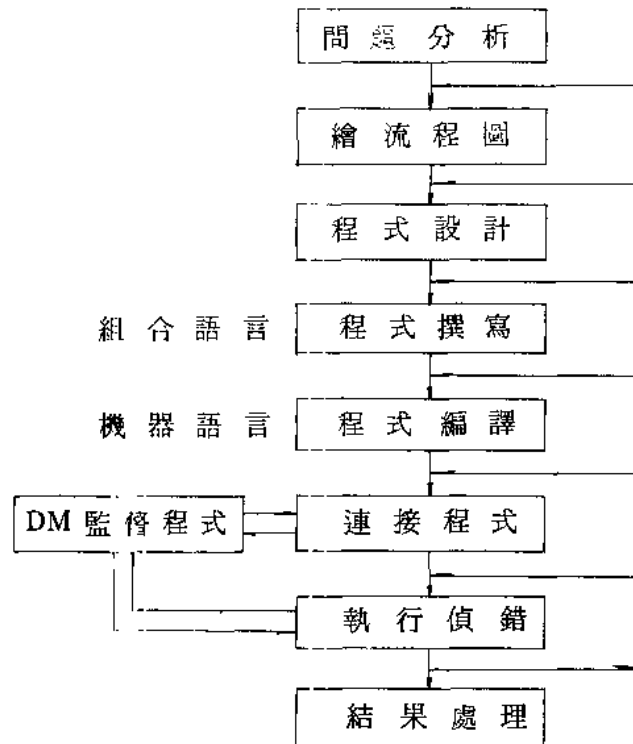
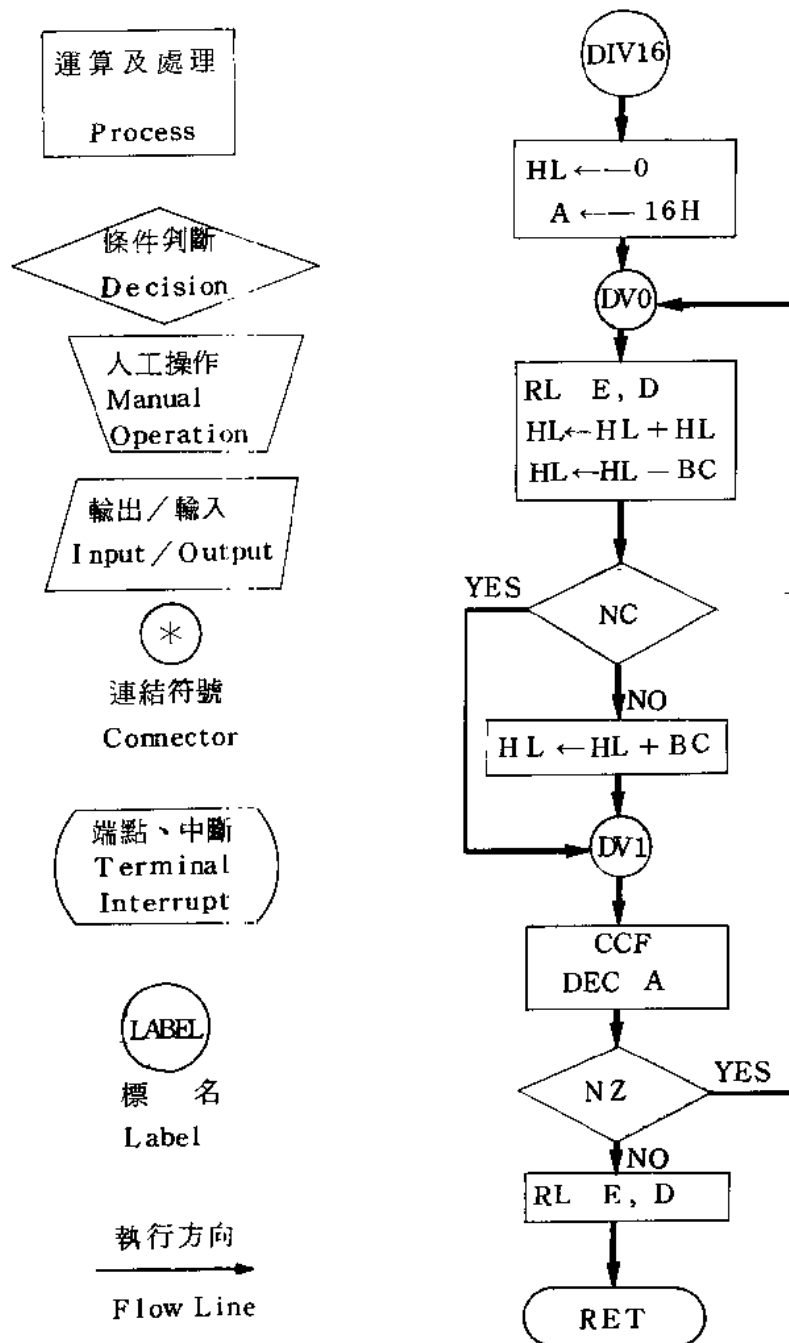


圖 一

二、繪製流程圖

流程圖能夠將一個繁複的程式以圖解的方式將其來龍去脈有條有理的表現出來，便利設計者做查核整理的工作，也可提供旁人對此程式的瞭解。流程圖的繪製可粗可細，粗至一個方塊符號代表一段副程式，細至一個符號僅表示一個指令，通常一個複雜的程式可先用粗略的系統流程圖來描述此程式的大綱，其次再就各部份作詳細的流程圖，流程圖最大的好處是能將程式執行的順序以箭線表示，並且用不同的符號表示程式在該階段所執行的性質，對於熟練的程式設計者，設計簡單的程式可能不必繪製流程圖即能著手程式設計之工作，圖二為流程圖使用之一般符號及範圍。



圖二 流程圖符號及 16BIT 除法範例

三、程式設計

程式的種類很多譬如簡單的數學運算方程式，輸入信號及輸出信號的轉換與產生程式，資料的編碼或解碼，週邊設備之驅動程式，或者繁複的編譯程式，監督程式，系統控制以及其他特殊應用之程式。每一類程式均有其特殊的設計技巧，勤練與充實是獲致成功的唯一途徑。

一般程式設計應考慮的事項如下：

- (1) 輸入信號或資料的獲取。
- (2) 輸出資料或信號的安排與產生。
- (3) 主程式之邏輯分析及算法。
- (4) 主程式與副程式之關係。
- (5) 暫存器之使用。
- (6) 主程式之記憶位址分配。
- (7) 副程式之記憶位址分配。
- (8) 數列資料之記憶位址分配及指標索引方式。
- (9) 程式初值及常數之設定。
- (10) 程式變數之設定。
- (11) 執行時間與時序的考慮。
- (12) 記憶容量大小的限制。
- (13) 資料之精確度與長度。
- (14) 是否有現成程式可資應用或參考。
- (15) 其他應特殊考慮事項。

四、程式撰寫

D、M的程式撰寫以組合語言為主，程式分成四大部份分別為標名（label），操作碼（Opcode），操作元（Operand）及說明（Comment）如下例所示：

標 名	操作碼與操作元	說 明
LABEL OPCODE & OPERAND COMMENT		
0001 DTB4	LD B,16	
0002 DB3	SRL H	
0003	RR L	
0004	RR D	
0005	RR E	;ROTATE HL DE RIGHT
0006	LD A,H	
0007	CALL DB1	
0008	LD H,A	;CORRECT H.
0009	LD A,L	
0010	CALL DB4	

```

0011      LD L,A      ;BINARY CORRECT L
0012      DJNZ DB3-$
0013      RET
0014 ;
0015 ;BINARY CORRECT ROUTINE
0016 DB4   BIT 7,A
0017      JR Z,DB1-$ ;IF BIT 7 OF A =1,SUB FROM 30H
0018      SUB 30H
0019 DB1   BIT 3,A
0020      JR Z,DB2-$ ;IF BIT 3 OF A =1,SUB FROM 03H
0021      SUB 3
0022 DB2   RET
0023      END

```

上例的程式若未加特別說明，不易看出其性質，若加上說明，則一望生義，對於繁複的程式而言，說明更屬重要，下面為程式加上程式名稱及功能說明後的結果：

從附加之說明事項可以明白的看出此段程式之功能，便於呼叫（CALL）及查核整理。

五、程式編譯組合

利用 PA 800 微型電腦系統之組合格式（Assembler）來編排組合我們的原始程式，其程序如下：

- (1) PA 800 POWER ON/RESET 進入 PA 800 DM 控制下。
- (2) 進入 EDITOR，開始編寫程式。
- (3) 程式編輯完畢，將原始程式交由 ASSEMBLER 編譯，若無錯誤，則交由 LINKER 將由 ASSEMBLER 編譯完成之 OBJECT PROGRAM LINK 到任何記憶位址去，否則，則再進入 EDITOR 並修改原始程式直到程式無誤為止。

下圖即為程式在 EDITOR 下編輯完之結果：

注意主程式最後一個指令為 JP 0E0BAH，此乃為檢查程式是否正確，令其跳到

```

0001 ; ** 4.DIGIT BCD TO BINARY CONVERSION ROUTINE **
0002 ;ENTRY :BCD DATA IN HL
0003 ;EXIT :BINARY DATA IN DE.
0004 ;REGISTER CHANGED :AF BC DE HL
0005 DTB4 LD B,16
0006 DB3   SRL H
0007      RR L
0008      RR D
0009      RR E      ;ROTATE HL DE RIGHT
0010      LD A,H
0011      CALL DB1
0012      LD H,A      ;CORRECT H
0013      LD A,L
0014      CALL DB4
0015      LD L,A      ;BINARY CORRECT L
0016      DJNZ DB3-$
0017      JP 0E0BAH

```

6 PA800 組合語言實習手冊

```

0018 ;
0019 ;BINARY CORRECT ROUTINE
0020 DB4 BIT 7,A
0021 JR Z,DB1-* ;IF BIT 7 OF A =1,SUB FROM 30H
0022 SUB 30H
0023 DB1 BIT 3,A
0024 JR Z,DB2-* ;IF BIT 3 OF A =1,SUB FROM 03H
0025 SUB 3
0026 DB2 RET

```

DM並顯示出 Register之內容，待程式完成後將其改為 RET 則此程式就可視為一個處理 4 DIGIT BCD TO BINARY CONVERSION 之 SUBROUTINE。

ADDR	OBJECT	ST #	
		0001	; ** 4 DIGIT BCD TO BINARY CONVERSION ROUTINE **
		0002	;ENTRY :BCD DATA IN HL
		0003	;EXIT :BINARY DATA IN DE.
		0004	;REGISTER CHANGED :AF BC DE HL
*0000	0610	0005	DTB4 LD B,16
*0002	CB3C	0006	DB3 SRL H
*0004	CB1D	0007	RR L
*0006	CB1A	0008	RR D
*0008	CB1B	0009	RR E ;ROTATE HL DE RIGHT
*000A	7C	0010	LD A,H
*000B	CD1F00	0011	CALL DB1
*000E	67	0012	LD H,A ;CORRECT H
*000F	7D	0013	LD A,L
*0010	CD1900	0014	CALL DB4
*0013	6F	0015	LD L,A ;BINARY CORRECT L
0014	10EC	0016	DJNZ DB3-
*0016	C3BAE0	0017	JP 0E0BAH
		0018	;
		0019	;BINARY CORRECT ROUTINE
*0019	CB7F	0020	DB4 BIT 7,A
001B	2802	0021	JR Z,DB1- ;IF BIT 7 OF A =1,SUB FROM 30H
*001D	D630	0022	SUB 30H
*001F	CB5F	0023	DB1 BIT 3,A
0021	2802	0024	JR Z,DB2- ;IF BIT 3 OF A =1,SUB FROM 03H
*0023	D603	0025	SUB 3
*0025	C9	0026	DB2 RET
		0027	END

ERRORS=0000

六、程式之輸入

此時 OBJECT PROGRAM 已經存放在 OBJECT AREA 中，我們將其 LINK 到 0 0 0 的地方，其方法如下：

- A (CR) ; 在 DM 控制下接入 A，A 表示 ASSEMBLY。
- OPTIONS ?K (CR) ; NO LISTING

* PASS1 END *

ERRORS = 0000

; ASSEMBLY 無誤

. L (CR)

; 回到DM後，按入LINK COMMAND “L”

; 從0 0 0 0開始。

. D 0 25 (CR)

0000 00 10 CB 3C CB 10 CB 10 ; 回到DM後，DISPLAY 0000 ~
0008 CB 18 7C 00 1F 00 67 7D 0025H中的資料。

0010 CB 19 00 6F 10 EC 03 BA ; MEMORY BLOCK
0018 E0 CB 7F 2B 02 06 10 CB

; DISPLAY結果發現MACHINE

; CODE 無誤。

由於本程式是將4 DIGIT BCD置於HL REGISTER中，而轉換成BINARY後，結果置於DE REGISTER中，因此首先看看REGISTER中之內容為何？

. R (CR)

PC AF BC DE HL IX I ; DISPLAY REGISTER CONTENT
51FF FFFF FFFF FFFF FFFF FFFF FF
5F A'F' B'C' D'E' H'L' 1Y 1F
FF90 FCB8 0056 3B00 0000 E8BB 00

我們可在DM控制下，設定HL一個值後，再讓程式執行之，當程式執行完畢後，跳回DM，並顯示出REGISTER值來，如下例，首先我們設HL為0100₍₁₆₎，則DE為0064₍₁₆₎。

七、程式之執行與偵錯

執行程式之第一步驟是設定程式之初值，程式執行後查核其結果，如果錯誤，則應用PA 800之偵錯程式（DEBUG）之功能分段執行，逐步查核修正，或者重新檢查定義及程式找出問題修正後，再重新執行之。

. R (CR)

PC AF BC DE HL IX I
027D 0054 00CF 0064 0000 FF29 E7
5F A'F' B'C' D'E' H'L' 1Y 1F
FF90 FCB8 0056 3B00 0000 E8BB 00
.D :1
:L 00 00
:H 00 01
:E 64 .
.G 0
PC AF BC DE HL IX I
027D 0054 00CF 0064 0000 FF29 E7
5F A'F' B'C' D'E' H'L' 1Y 1F
FF90 FCB8 0056 3B00 0000 E8BB 00

; DM控制下顯示出REGISTER
之CONTENT。

; 顯示出L REGISTER之CONTENT

; 設定HL為0100₍₁₆₎

; 執行結果DE為0064。

5 PA800 組合語言實習手冊

```
.D':L
:L 00 34
:H 00 12
:E 64
.G 0
PC AF BC DE HL IX I
027D 0054 00CF 04D2 0000 FF29 E7
SP A'F' B'C' D'E' H'L' IY IF
FF90 FCB8 0056 3B00 0000 E8BB 00
```

；設定 HL 爲 1234₍₁₆₎
；執行結果 DE 爲 04D2₍₁₆₎

八、程式完成

程式執行無誤後應留下正確的書面程式 (Program LIST)，並給予適當之命名，錄入磁帶中以資運用，或者加以簡化，以節省記憶組佔用之位址，或加以修正以節省執行之時間。

實習一

PA800-DM開機與基本試驗

主 旨：1 瞭解PA800微型電腦使用DM操作系統之開機與試驗方法。
2 熟悉資料傳送練習。

需用器材：PA800 微型電腦。

相關知識：

PA800 為適合教學與產品應用之微電腦系統。採用DM操作程式之系統組合方法如圖 1 - 1 所示，包括主機，電視螢幕，英文鍵盤，卡式磁帶機，與英文印字機等。

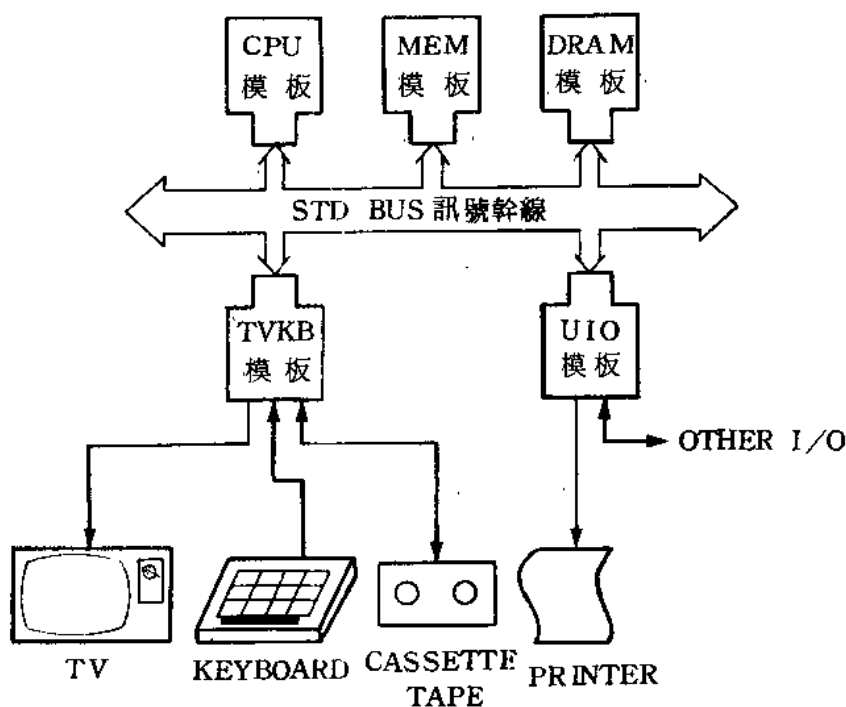


圖 1 - 1

1 主機：

PA800 主機為整個微型電腦之中心。包括容納 10 片模板之母板槽，電源供應器，冷却風扇，面板 RESET 按鈕，背板連接器，指示燈等，標準 DM 操作系統之模板組合如下：

- (1) CPU 1 模板：Z80 CPU, CTC, 4K EPROM, 1K RAM。
- (2) MEM 1 模板：12K EPROM, 4K RAM。