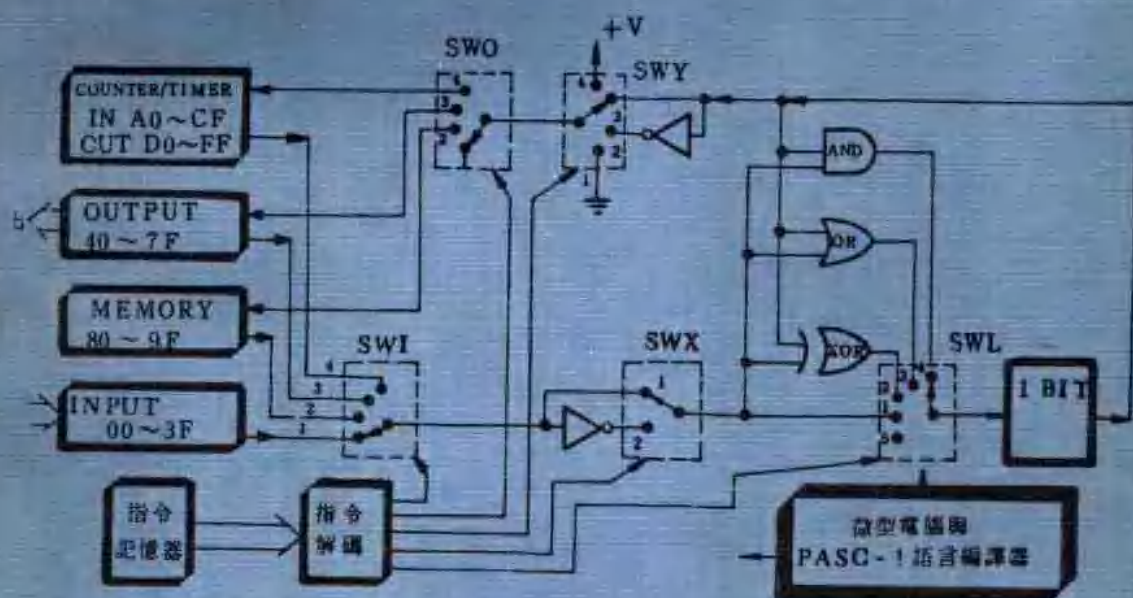




PASC-1

順序控制語言

0 1 2 3 4 5 6 7 8 9 A B C D
E F out J P Q P A R



全亞電子工業股份有限公司編寫
全華科技圖書公司印行

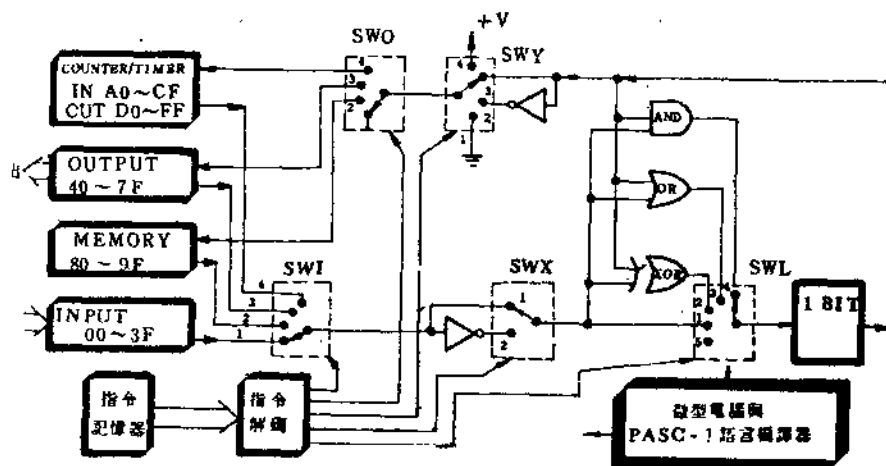


PASC-1

順序控制語言

0 1 2 3 4 5 6 7 8 9 A B C D

E F out JPO PAR



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

PASC-1

順序控制語言

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・7083595

台中辦事處：台中市精誠五街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・564-1819

581-1362・581-1347

郵政特准掛號：100B36

發行者 陳本源

印刷者 欣瑜彩色印刷廠

定價 新臺幣 66 元

再版 中華民國72年3月

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

PASC-1 順序控制語言

PAN-ASIA SEQUENCE CONTROL LANGUAGE VERTION-ONE

一、前 言	1
二、PASC-1 指令陳述方式	2
三、PASC-1 程式範例	4
四、PASC-1 位址分配	7
五、PASC-1 指令說明	9
六、PASC-1 順序控制系統邏輯結構	18
七、階梯式電機控制圖處理	19
八、流程式順序控制圖處理	21
九、鍵盤與顯示幕使用	25
十、鍵盤說明	36
十一、PASC-1 編譯器工作狀態	38
十二、COUNTER/TIMER 使用	41
十三、簡易階梯控制圖及流程圖實例	45
十四、流程圖控制實例	59
十五、階梯控制圖實例	76

PASC-1 順序控制語言

一、前 言

科技日益精進，工業控制技術不斷地改良。今天要求控制器材程式化（PROGRAMMABLE），以便控制方式能隨著技術之更新而隨時改變，已為工業界所迫切需要。全亞電子公司為配合此種潮流，特推出PASC-DT型工業用微型電腦，並提出PASC-1（PAN-ASIA SEQUENCE CONTROL LANGUAGE-1）順序控制語言。使控制功能藉口語化之指令陳述而完成。該微型電腦系統除PASC-1語言編譯器外，並提供一系列之操作命令鍵，使設計或操作人員能方便地輸入、儲存、試驗其控制程式，並協助程式偵錯。

PASC-1之程式指令可在7節式顯示幕（7-SEGMENT DISPLAY）上顯示出來。操作人員不需在電傳打字機或電視螢幕終端機前，即可用口語化之指令來處理控制程式。故處理PASC-1順序控制語言不需昂貴之機器。通常其價格遠低於電傳打字機。使用者可於PASC-DT微型電腦上發展PASC-1順序控制語言，透過鍵盤輸入，螢幕顯示及特殊編輯功能，編寫程式。利用PASC-DT上之I/O模擬器，模擬現場控制狀況，易學、易用，是一發展PASC-1順序控制語言的利器。

PASC-1控制語言可處理64輸入點，64輸出點，32記憶點與16COUNTER或TIMER之ON/OFF順序控制問題。指令之設計兼顧傳統之控制方式與現代之控制觀念。既有之控制系統，不論是以時間圖（TIMING CHART）、流程圖（FLOW CHART）、階梯圖（LADDER DIAGRAM），邏輯符號圖甚或以布林（BOOLEAN）邏輯式表示，均能迅速轉換成PASC-1程式語言。傳統之程序設計工程師可在一天之講習中，瞭解PASC-1語言之要點，進而實際用來處理順序控制問題。

二、PASC-1 指令陳述方式

PASC-1 語言之指令陳述分為步驟 (STEP)、操作名稱 (COMMAND)、邏輯極性 (POLARITY)、操作對象 (OPERAND) 與說明 (COMMENT) 等五部份。說明部份只顯示在文件資料上，在實際機器上操作時，7 節式顯示幕只顯示指令之前四部份。

下列三指令表示第 46 號輸出端必須在第 01 號輸入端 “AND” 第 02 號輸入端均為 1 (或 ON) 時，才輸出 0 (或 OFF)，否則輸出 1。茲以此三指令為例說明 PASC-1 指令之陳述方式。

```

12      IN      01 ; 輸入 (IN) 第 01 號輸入端之訊號。
13      AND      02 ; 將其與第 02 號輸入訊號作 “AND” 判斷。
14      OUT  —  46 ; 將結果之反態輸出 (OUT) 到第 46 號輸出端。

```

上例最後之指令包括指令之五部份。其中最左方之 “14” 稱為步驟 (STEP)，意指此指令在整個程式中排列在第 14 步驟 (十六進制)。其次 “OUT” 是指令名稱 (COMMAND)，指定要做輸出之動作。再右方 “—” 記號代表邏輯之極性，表示要用相反之邏輯狀態來處理 (輸出)。“—” 之右方 “46” 是此指令之操作對象，指示輸出之對象是第 46 號輸出端。最右方 “;” 記號後面是指令之說明或註解，此部份在文件整理時相當重要，可增加程式之可讀性。

PASC-1 程式中有關數字部份均以十六進制數字方式表示。事實上十六進制數字是 4 個二進制數字之濃縮。幾乎目前各種大小型以至微型電腦都用十六進制數字來表示其基本操作與演算。讀者熟悉十六進制之演算，處在此電腦時代將是一項十分有用之技能。

PASC-1 之指令形態如下，茲將各部份說明之：

NN PPPSXX ; CCG

例：14 OUT — 46 ; 輸出相反之訊號到第 46 號輸出端。

- (1) NN : 在 PASC-1 中，NN 為兩位數之十六進制數字。代表第幾步驟或第幾條指令。故 PASC-1 之步驟為 00 ~ FF，相當於十進制之 256。(進一步之 PASC 語言，步驟可達三位數字，則可處理更大更複雜之順序控制問題)。

- (2) PPP : 以助記縮語 (NMEMONIC) 來表示之操作動作, 即為指令 (COMMAND) 名稱。通常為 2~3 個字母。前例中“IN”代表 INPUT, “AND”代表邏輯 AND 判斷。“OUT”代表 OUTPUT 等動作。
- PASC-1 之助記縮語 (或指令名稱) 有 20 種 (十進制), 代表 20 種不同動作之指令。詳見指令說明部份。
- (3) S : 代表控制動作之正反狀態。S 部份可為空白, 如前例之第 12、13 步驟, 表示正常直接之邏輯動作。若 S 位置為“—”高橫綫, 則表示要用相反的邏輯狀態操作。PASC-1 之指令中只有五個指令後面可加“—”高橫綫符號, 它們是 IN、OUT、AND、OR、CP。
- (4) XX : 代表操作對象, 亦為二位數之十六進制數字 (進一步之 PASC 語言可擴展到三位數)。不同之指令, XX 代表不同之意義, 某些指令無操作對象, 則 XX 位置為空白。
- IN、OUT、AND、OR、CP 等指令, 其 XX 部份代表各種內外端點之號碼, 或稱位址 (ADDRESS)。
 - JP、JPO、JP1、REP、CAL、ETR、DTR 等指令, 其 XX 部份代表程式之步驟數。例如 JP 53 表示跳到第 53 步驟去執行。
 - SCT 指令之 XX, 代表 COUNTER/TIMER 之編號, 或基本時間單位。
 - PAR 指令之 XX, 則代表時間、次數、計數位址等數據。
- (5) CCC : 此部份為程式之加註說明, 僅表示在文件上。不輸入到機器去執行。控制程式應儘量加註說明。增加程式之可讀性, 非但使別人容易瞭解。在程式試驗偵錯時可幫助自己把錯誤減少到最少。

三、PASC-1 程式範例

PASC-1 之控制程式簡單易學。縱未看過各種指令說明，通常亦大都能瞭解程式所指之意義。下面為 PASC-1 之程式範例，讀者若有不瞭解之處，請暫時跳過。俟閱讀後面相關資料後，再來看此範例必能得心應手。

【例 1】 第 01 號與 02 號輸入端分別為高低液位警戒開關。液位高於其裝置位置時輸入 1，否則輸入 0。第 40、41、42 號輸出端控制紅綠藍三個指示燈，第 43 號接警鈴。控制之要求如下：

- a. 液位太高：紅燈(40)亮，警鈴(43)響。
- b. 液位正常：綠燈(41)亮。
- c. 液位太低：藍燈(42)亮。警鈴(43)響。
- d. 液位開關同時指示太高與太低：紅藍燈同時亮，警鈴每 2 秒鐘響一次，表示故障。

【解】 00 SCT 11 ; SET COUNTER/TIMER。指定第 1 號作為
; TIMER 之用。採用第 1 種基準時間單位，亦即
; 每一時間單位為 1 秒。

01 PAR 02 ; 計時為 2 個時間單位。

02 IN 01 ; 輸入第 01 號高液位警戒開關之訊號。

03 OUT 40 ; 將其引接到第 40 號紅燈控制端。

04 IN 02 ; 輸入第 02 號低液位警戒開關之相反訊號。

05 OUT 42 ; 將其引接到第 42 號藍燈控制端。

06 AND 40 ; AND 判斷第 42 號與第 40 號端點是否均為 1。

07 JP1 0D ; 若兩者均為 1，則跳到第 0D 步驟去處理警鈴間
; 歇響之程式。否則繼續下一個指令。

08 IN 40 ; 輸入第 40 號紅燈控制端點之訊號。

09 OR 42 ; OR 判斷第 40 號與第 42 號是否有任何一個為 1。

0A OUT 43 ; 將 OR 判斷之結果輸出到第 43 號警鈴。

0B OUT 41 ; 將判斷結果之反態輸出控制第 41 號綠燈。

0C JP 10 ; 無條件跳到第 10 步驟去繼續程式。

0D SET A1 ; 開動第 1 號 TIMER，若 TIMER 已開動則這個
; 指令無效。

0E IN F1 ; 引入 TIMER 之 0/1 交變輸出端之訊號，每一計
; 時單位（1 秒）交變一次。

0F OUT 43 ; 轉輸出到第 43 號警鈴。
 10 JP 02 ; 跳到第 02 步驟反覆不斷地執行。

【例 2】 第 01 與 02 號輸入端均為按鈕，按下時輸入 1。第 40 與 41 號分別控制馬達之 ON / OFF 與高 / 低速運轉。控制之要求如下：

- a. 第 01 號按鈕按下時，馬達低速運轉，4 秒鐘後自動轉為高速，運轉 30 分鐘後停止。
- b. 運轉時，若第 02 號按鈕按下則馬達立即停止。

操作程式如下：

00 ETR 02 ; ENABLE TIME SHARE ROUTINE。指
 ; 定第 02 步驟之副程式為分時執行副程式。以後
 ; 每 1 / 60 秒即自動分配一部份時間去執行這個被
 ; 指定之副程式。
 01 JP 06 ; 跳過分時執行副程式。
 ; * 以下第 02 ~ 05 步驟為分時執行副程式。
 02 IN 02 ; 輸入第 02 號按鈕之狀態。
 03 JPO 05 ; 若輸入為 0，則跳到第 05 步驟結束分時執行副
 ; 程式。否則做下一個指令。
 04 RES 40 ; RESET 第 40 號輸出端點為 0，馬達停止。
 05 RET ; RETURN，回到原程式。結束分時程式。
 06 RES 41 ; RESET 41 號輸出端為 0，預置為低速情況。
 07 IN 01 ; 輸入 01 號按鈕之情況。
 08 JPO 07 ; 若輸入 0，則表示按鈕未按。跳到第 07 步驟重
 ; 新輸入，等待按鈕按下。
 09 SET 40 ; 執行到此指令表示第 01 號按鈕已按下，故 SET
 ; 第 40 號輸出為 1，馬達啟動（低速）。
 0A DLY 04 ; DELAY，延遲 4 個時間單位。
 0B PAR 0A ; PARAMETER，DLY 指令後之 PAR 指定每
 ; 一時間單位應乘以幾個 0.1 秒。故此 DLY 指令
 ; 延時 4 秒。（0A 相當於十進制 10）。
 0C SET 41 ; SET 第 41 號為 1，馬達變為高速運轉。
 0D DLY B4 ; 延遲 180 個時間單位（十進制）。
 0E PAR 64 ; 每個時間單位為 0.1 秒乘以 100（十進制）；亦

```

                                ; 即延遲 1800 秒，或 30 分鐘。
0F RES 40 ; RESET 40 號輸出端為 0，停止馬達。
10 DTR 02 ; DISABLE TIME-SHARE ROUTINE，
    :      ; 停止第 02 步驟之分時執行副程式。
11 :      ; 繼續其他程式。
    :

```

【例 3】第 04 號輸入端為感光偵測器，輸送機主軸每旋轉一週遮光一次，因而產生 $0 \rightarrow 1 \rightarrow 0$ 之變化。若主軸轉一週前進 1 公分。第 47 號輸出端控制馬達之 ON/OFF。則下列程式使輸送機啟動前進一公尺後停止。

```

00 SCT 03 ; 指定第 3 號 COUNTER/TIME 作為 COUNTER
    :      ; 用。
01 PAR 64 ; 第 1 個 PARAMETER 指定計數預調值為 100 (
    :      ; 十進制)。
02 PAR 04 ; 第 2 個 PARAMETER 指定計數輸入端為第 04
    :      ; 號輸入端。
03 RES B3 ; RESET 第 B3 號端點，亦即 RESET 第 3 號
    :      ; COUNTER。
04 SET A3 ; SET 第 A3 號端點，亦即開動第 3 號 COUNTER。
05 SET 47 ; SET 第 47 號輸出端點為 1，開動馬達。
06 IN E3 ; 輸入第 E3 端點訊號，亦即 COUNTER 狀態訊
    :      ; 號。
07 JPO 06 ; 若輸入 0，表示 COUNTER 計數未完全，重新
    :      ; 到第 06 步驟去輸入，一直到第 E3 端輸入 1 為
    :      ; 止（表示計數完畢）。
08 RES 47 ; RESET 第 47 號端點為 0，馬達停止。
09 ;      ; 繼續下面之程式。

```

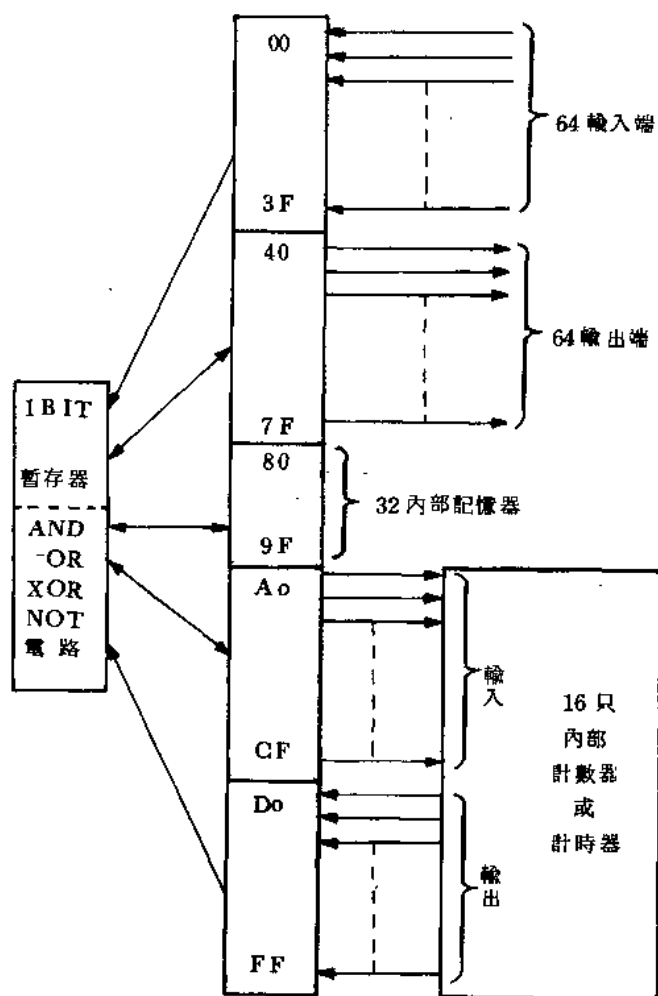
四、PASC-1 位址分配

PASC-1 控制程式之指令中，對於每一操作端點均予以編號，亦即每一個操作端點分配一個位址。PASC-1 之位址為兩位數之十六進制數字（進一步之 PASC 語言將採 3 位數字）。故位址總共有 256（十進制）個。位址之分配使用如下：

1. 00 ~ 3F : 編為輸入端點，總共有 64 個號碼（十進制）。輸入端點之訊號只能輸入，不能加以更改。
2. 40 ~ 7F : 編為輸出端點，總共有 64 個號碼（十進制）。輸出端點之訊號可以重新輸入。故未實際接線到外部使用之位址可作為記憶端點用。
3. 80 ~ 9F : 編為記憶端點，總共有 32 個號碼（十進制）。程式可輸出資料到記憶端點，亦可隨時自記憶端點輸入資料。記憶端點無實際之接線，不能接到外部使用。
4. A0 ~ CF : 編為 COUNTER 或 TIMER 之控制端點。分 A、B、C 三組控制 16 個（十進制）COUNTER / TIMER 之動作。右方之數字代表 COUNTER / TIMER 之編號。
 - a. A0 ~ AF : 控制 COUNTER / TIMER 之啟動（START）。
 - b. B0 ~ BF : 作為 COUNTER / TIMER 之 RESET 控制。
 - c. C0 ~ CF : 作為 TIMER 時無作用，可作記憶端點用。
 作為 COUNTER 時，若程式不另行指定計數輸入端點，則此端點作為計數脈波之輸入端點。
 AX, BX, CX (X 代表 0 ~ F 之 COUNTER / TIMER 編號) 端點均可由程式隨時輸入或輸出訊號。故未使用之 COUNTER / TIMER 其對應之 AX, BX, CX 端點可作為記憶端點用。
5. D0 ~ FF : 編為 COUNTER 或 TIMER 之輸出端點。反應 COUNTER / TIMER 之狀態。此端點同 INPUT 端點一樣只能被程式輸入使用，程式輸出資料到此端點為不合法之指令。
 - a. D0 ~ DF : 若呈現 1，表示 TIMER 正在計時，或 COUNTER 正在計數。
 - b. E0 ~ EF : 作 TIMER 用時，輸出 ON - DELAY 訊號。
 作 COUNTER 時，若輸出 1，表示 COUNTER 計數完成。
 - c. F0 ~ FF : 作 TIMER 用時，在 TIMER 開動期間，此端點為 0 / 1 反

8 PASC-1 順序控制語言

覆交變之訊號。其半週期為該TIMER之基本計時單位。
作COUNTER用時FO~FF為無意義之訊號。



圖(4-1) PASC-1 操作位址分配

五、PASC-1 指令說明

PASC-1 有 20 個指令，分別用 20 個助記縮語 (NMEMORIC) 表示指令名稱。各種訊號以內部 1 BIT 之暫存記錄器 (1 BIT REGISTER 簡稱 R) 為中心而處理。茲將各指令之動作說明如下：

1. IN XX (in) INPUT TO REGISTER

將第 XX 位址之邏輯狀態 (0 或 1) 輸入到內部暫存器。XX 為 00 ~ FF 之任何位址。若 XX 之左方有“—”高橫線記號，則表示用相反的邏輯狀態輸入。

例：

IN 05 ; 第 05 號輸入端點之訊號輸入暫存器。

IN —48 ; 第 48 號輸出端訊號之反態輸入暫存器。

IN 94 ; 第 94 號記憶器之訊號輸入暫存器。

IN D4 ; 第 D4 號端點，亦即呈現第 4 號 COUNTER / TIMER 動作狀態之端點訊號輸入暫存器。

OUT XX (out) OUTPUT TO XX

將暫存器之邏輯狀態輸出到第 XX 號位址。XX 為 40 ~ CF 之十之進制數字。若 XX = 00 ~ 3F 或 XX = D0 ~ FF，則為不合法之指令。若 XX 之左方有“—”記號，則將暫存器之相反狀態輸出。

例：

OUT 49 ; 將暫存器之訊號輸送到第 49 號輸出端。

OUT 05 ; 不合法之指令，因為 05 為輸入端。

OUT —83 ; 將暫存器之反態輸送到第 83 號記憶器。

OUT A3 ; 將暫存器之訊號輸送到第 A3 位址，亦即第 3 號 COUNTER / TIMER 之 START 端。

3. AND XX (and) LOGICAL AND

將 XX 位址之訊號與 REGISTER 作邏輯 AND 判斷。判斷結果存回 REGISTER。XX 為 00 ~ FF 之十六進制數字。XX 之前若加“—”高橫線記號

10 PASC-1 順序控制語言

，則取XX位址之反態作AND判斷。

例：

AND 03 ; 將REGISTER與第03號輸入端之訊號作AND判斷，結果存回REGISTER。

AND—80 ; 將REGISTER與第80號記憶器之反態訊號作AND判斷；，結果存回REGISTER。

例：下列程式將第01，02，03號輸入端作邏輯AND判斷，結果輸出到第41號輸出端。

```
15 IN 01 ; R ← 01
16 AND 02 ; R ← 01 · 02
17 AND 03 ; R ← 01 · 02 · 03
18 OUT 41 ; 41 ← R ← 01 · 02 · 03
```

4. OR XX (OR) LOGICAL OR

將XX位址之訊號與REGISTER作邏輯OR判斷。判斷結果存回REGISTER。XX為00～FF之十六進制數字。若XX之前加“—”高橫線，則取XX之反態來作判斷。

例：

OR 54 ; 將REGISTER與第54號輸出端之訊號作邏輯OR判斷，結果存回REGISTER。

OR—24 ; 將REGISTER與第24號輸入端之反態訊號作邏輯OR判斷；，結果存回REGISTER。

例：下列程式將第02，03號輸入端之正態訊號與第13號輸入端之反態訊號作邏輯OR判斷。結果輸送到第98號記憶器儲存起來。

```
20 IN 02 ; R ← 02
21 OR 03 ; R ← 02 + 03
22 OR—13 ; R ← 02 + 03 +  $\overline{13}$ 
23 OUT 98 ; 98 ← R ← 02 + 03 +  $\overline{13}$ 
```

5. CP XX (CP) COMPARE WITH XX

將REGISTER之訊號與XX位址之訊號比較，若相同則置REGISTER為0，否則置REGISTER為1。XX為00～FF之十六進制數字。若XX之左方有“—”高橫線，則取XX位址之反態訊號來作比較。

PASC-1 每次處理1 BIT，故CP指令事實上與邏輯XOR (EXCLUS-

IVE OR) 判斷相同，故若控制程式需作 XOR 判斷，則可用 CP 指令。

例：

CP 03 ; 將 REGISTER 與第 03 號輸出端比較，若相同則 $R \leftarrow 0$ ，不同則 $R \leftarrow 1$ 。

CP 80 ; 將 REGISTER 與第 80 號記憶器比較，若相同則 $R \leftarrow 1$ ，不同則 $R \leftarrow 0$ (與上一指令相反)。

CP 40 ; 將 REGISTER 與第 04 號輸入端作邏輯 EXCLUSIVE OR 判斷，結果存入 REGISTER。

6. SET XX (555) SET TO LOGICAL ONE

將第 XX 號位址之訊號設定為 1。XX = 40 ~ CF 之十六進制數字。若 XX = 00 ~ 3F 或 D0 ~ FF，則為不合法之指令。若 XX = A0 ~ AF，亦即 COUNTER/TIMER START 端之位址，而 COUNTER/TIMER 未開動，則 SET 指令啟動 COUNTER/TIMER，不論 A0 ~ AF 內之訊號原來為 0 或為 1。若 COUNTER/TIMER 已開動，則本指令僅 SET 該端點為 1。若計數或計時結束，則 SET 重新啟動 COUNTER/TIMER。

例：

SET 44 ; 將第 44 號輸出端設為 1。

SET A3 ; 啟動第 3 號 COUNTER/TIMER。

SET E4 ; 不合法之指令。(E4 為 COUNTER/TIMER 之輸出端)。

7. RES XX (777) RESET TO LOGICAL ZERO

將第 XX 號位址之訊號重設為 0。XX = 40 ~ CF 之十六進制數字。若 XX = 00 ~ 3F 或 D0 ~ FF，則為不合法之指令。若 XX = B0 ~ BF，則 RESET COUNTER/TIMER，不論 XX 之內容原來為 1 或為 0。

例：

RES 40 ; 將第 40 號輸出端 RESET 為 0。

RES 54 ; 自第 54 號輸出端輸出 0。

RES B9 ; RESET 第 9 號 COUNTER/TIMER。

RES 32 ; 不合法之指令，因 32 為輸入端。

12 PASC-1 順序控制語言

8 JP XX (JP) UN-CONDITIONAL JUMP

無條件跳到第XX步驟去執行程式。XX = 00 ~ FF之十六進制步驟。PASC-1之程式通常需以JP指令作為結尾。

例：

JP 5A ; 跳到第5A步驟去執行程式。

JP 00 ; 程式結束，跳到第00步驟重新執行程式。

9 JP0 XX (JP0) JUMP IF ZERO

若暫存器之狀態為0，則跳到第XX步驟去執行，否則繼續下面之程式。XX = 00 ~ FF之十六進制步驟。

例：

10 IN 01 ; 輸入第01號輸入端訊號。

11 AND 02 ; 與第02號輸入端作AND判斷，結果存於暫存器。

12 JP0 10 ; 若暫存器為0，則跳回第10步驟繼續作判斷，否則執行下面第13步驟程式。此三指令在等待第01與第02輸入端同時出現1。

13 :

10 JP1 XX (JP1) JUMP IF ONE

若暫存器為1，則跳到第XX步驟去執行程式，否則繼續下面之程式。XX = 00 ~ FF之十六進制步驟。

例：

10 IN 13 ; 輸入第13號輸入端之訊號至暫存器。

11 JP1 24 ; 若暫存器為1，則跳到第24步驟。否則繼續執行12步驟之程式。

12 :

11 CAL XX (CAL) CALL SUBROUTINE

呼叫第XX步驟之副程式。XX = 00 ~ FF之十六進制數字。PASC-1程式執行到CAL指令時，將下一個指令之步驟數記入堆疊記憶區(STACK)，而後跳到第XX步驟去執行程式。堆疊記憶區可記入16個步驟數，最早記入