

GOTOP

# PLD

## 數位邏輯 實務應用

碁峯資訊

陳兆麟

## 30分鐘系列

易懂・易學・易用



書號: CQ001  
MS-DOS



書號: CQ002  
倚天中文系統



書號: CQ003  
PEI I/PEI II



書號: CQ004  
LOTUS 123



書號: CQ005  
DBASE III PLUS



書號: CQ006  
電腦基本概論



書號: CQ007  
硬碟管理手冊



書號: CQ008  
電腦維修手冊



書號: CQ009  
電腦病毒防治



書號: CQ010  
記憶體管理



書號: CQ011  
Windows 中文版入門



書號: CQ012  
Windows 中文版進階



書號: CQ013  
Excel 中文版



書號: CQ014  
Word 中文版



書號: CQ015  
PowerPoint



書號: CQ016  
漢書



雙色印刷

30分鐘系列套書  
15本精美包裝  
一套2400元

精巧的 32開 - 帶得輕鬆方便  
輕薄的208頁 - 看得輕鬆愉快  
超值的160元 - 買得滿心歡喜

請至各大書局、資訊廣場購買或郵政劃撥  
基峰資訊股份有限公司 帳號: 14244383

## 輕鬆學系列

學得輕鬆 • 用得輕鬆



書號: CV001  
漢書



書號: CV002  
Windows  
中文版



書號: CV003  
Lotus 易備算



書號: CV004  
Corel Draw



書號: CV005  
Foxpro  
for Windows



書號: CV006  
MS-DOS



書號: CV007  
Word 中文版



書號: CV008  
Excel 中文版



書號: CV009  
Power Point



書號: CV010  
Turbo C/C++



雙色印刷

每本280元 - 讓您買得輕鬆

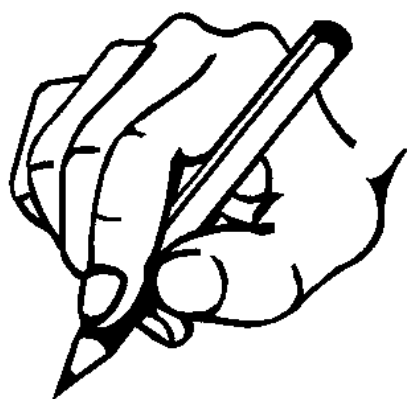
美編創意 - 讓您看得輕鬆

範例練習 - 讓您學得輕鬆

重點複習 - 讓您用得輕鬆

請至各大書局、資訊廣場購買或郵政劃撥  
基峰資訊股份有限公司 帳號: 14244383

附



B

錄

---

# **PALASM 4 的 語法介紹**

(1) .CLKF (用來觸發暫存器 CP 腳)

例如：

```

1      CLK
2      A          ;INPUT
3      B          ;INPUT
13     AREG       ;OUTPUT

```

-----

```

AREG = A + B
AREG.CLKF=CLK

```

這個意思是說當輸入腳 CLK 觸發後，輸出腳 AREG 之暫存器 CP 亦被觸發此時  $A + B$  將輸出至 AREG，如圖 B-1。

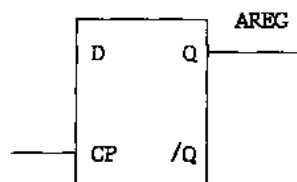


圖 B-1

(2) CLOCK SOURCE (一系統若有數個 CLOCK 可用來指定某一特定的 CLOCK)

作為目前 CLOCK

CLKF = CLOCK SIGNAL

(3) CLKF (用來定義某一個 CLOCK 輸入作為狀態圖所需的 CLOCK)  
在系統中也許有數個 CLOCK 輸入我們可指定特定的 CLOCK 作為 STATE MACHINE 所需同步之 CLOCK。

例如：

```
CLOCKF = CLK1
```

輸入 CLK1 作為 STATE MACHINE 之 CLOCK。

- (4) CLOCKF (在指定的輸入腳輸入 — CLOCK)

```
SETF /CLOCK  
CLOCKF CLCOK
```

設定 CLOCK 輸入腳為低電位，然後在 CLOCK 腳產生一 。

- (5) CHECK (檢查模擬後的結果是否正確)

```
CLOCKF CLOCK  
CHECK A /B C
```

在 CLCOK 腳輸入一正緣脈波，然後檢查是否 A=1，B=0，C=1 若不是則會產生 錯誤訊息。

- (6) CHECKQ (檢查模擬後的暫存器輸出結果是否正確)

例如：

```
CHECKQ Q0 /Q1 PLAY
```

檢查 Q0 是否為 1，Q1 是否為 0，PLAY 是否為 1。



# (7) CASE (條件方程式)

例如：

CASE ( A , D )

BEGIN

0: BEGIN

$X = Y * Z$

END

3: BEGIN

$X = Y + Z$

END

2: BEGIN

$X = /Y * Z$

END

OTHERWISE: BEGIN

$X = Y * /Z$

END

當  $A=0$  ,  $D=0$  滿足第一個條件 執行  $X = Y * Z$

當  $A=1$  ,  $D=1$  滿足第二個條件 執行  $X = Y + Z$

當  $A=1$  ,  $D=0$  滿足第三個條件 執行  $X = /Y * Z$

其它狀態執行  $X = Y * /Z$

# (8) CONDITIONS (在轉移方程式中設定輸入條件)

S1:- C1 -> S2

+ C2 -> S3

CONDITIONS

C1=A \* B

C2=A \* C

當輸入符合 C1 時狀態由 S1 轉移至 S2

當輸入符合 C2 時狀態由 S1 轉移至 S3

(9) DEFAULT\_BRANCH (共用預設跳躍)

在轉移方程式中若所有輸入都不符合則跳至預設狀態。

例如：

```
DEFAULT_BRANCH HOLD_STATE
```

轉移方程式中若所有輸入都不符合則保持在原狀態

```
DEFAULT_BRANCH NEXT_STATE
```

轉移方程式中若所有輸入都不符合則跳至 PDS 檔中所列之下一狀態。

(10) DEFAULT\_OUTPUT (設定暫存器之輸出值)

例如：

```
DEFAULT_OUTPUT /OUT2
```

設定 OUT2 之輸出為 0

(11) FOR ... TO DO (迴圈;重覆執行一連串的指令)

例如：

```
TRACE_ON SET /OE CLOCK /OE A B C
SETF /SET OE /CLOCK
FOR I:-1 TO 20 DO
  BEGIN
    CLOCKF CLOCK
    CHECK A B C
  END
TRACE_OFF
```

CLOCK 腳位連續輸入 20 個脈波



- (12) GROUP (將一群腳位定義成相同的信號名稱)

例如：

```
PIN 2 X REG
PIN 3 Y REG
PIN 4 Z REG
PIN 5 W REG
GROUP QRST X Y Z W
```

QRST.RSTF = RES

在組譯時擴展為

```
X.RSTF = RES
Y.RSTF = RES
Z.RSTF = RES
W.RSTF = RES
```

- (13) IF .. THEN ..ELSE (當條件成立則執行某一指令否則執行另一指令)

```
TRACE_ON SET /OE CLOCK /OE A B C
SETF /SET OE /CLOCK
IF ( A * /B * /C )
  BEGIN
    CLOCKF CLOCK
  END
ELSE
  SETF SET
TRACE_OFF
```

當  $(A * /B * /C = 1)$ ，自 CLCOK 腳輸入一脈波，否則設定 SET=1

(14) LOCAL DEFAULT BRANCH (局部預設跳躍)

對某一轉移方程式中若所有輸入都不符合則跳至預設狀態。

```
PRESENT STATE:-CONDITION NAME -> NEXT STATE
      + CONDITION NAME -> NEXT STATE
      .....
      +STATE NAME
```

若所有輸入都不符合跳至 STATE NAME。

(15) MINIMIZE\_OFF, MINIMIZE\_ON (指定某些布林方程式在組譯時不作)

例如：

```
MINIMIZE_OFF
OUT1 = A * B * C
MINIMIZE_ON
```

OUT1 = A \* B \* C，這一方程式在組譯時不作減化動作，之後的方程式可作減化動作。

(16) .OUTF (用來定義 MOORE 或 KEALY MACHINE 每個狀態的輸出)

例如：

```
MEALY MACHINE

S1.OUTF= CONDITION1 -> OUT1
      + CONDITION2 -> OUT2
      + CONDITION3 -> OUT3
```



當 S1 之輸入為 CONDITION1 時輸出為 OUT1

當 S1 之輸入為 CONDITION2 時輸出為 OUT2

當 S1 之輸入為 CONDITION3 時輸出為 OUT3

例如：

MOORE MACHINE

TWO.OUTPUT = /A1 \* /A2 \* /A3

TWO 狀態之輸出為 /A1 \* /A2 \* /A3

#### (17) PRELOAD 設定暫存器的初值

例如：

PRELOAD    /A B   /C

設定暫存器 A 輸出為 0

設定暫存器 B 輸出為 1

設定暫存器 C 輸出為 0

#### (18) PRLDF (在模擬段設定輸出暫存器之值；僅適用於

PAL22IP6, PAL23S8, PAL16RA10, PAL20RA10)

SIMULATION

PRLDF X1 /X2

設定 X1 輸出暫存器為 1

設定 X2 輸出暫存器為 0

## (19) REGISTERED (用來定義輸出腳為暫存器輸出)

例如：

```
PIN      15      OUT1      REGISTERED
```

OUT1 為具有暫存器之輸出。

## (20) .RSTF .SETF (用來設定暫存器輸出腳位為 LOW 或 為 HIGH)

例如：

```
PIN  2      SET      ;INPUT
PIN  3      RST      ;INPUT
PIN  12     AREG      ;OUTPUT
AREG = A + B
AREG.SETF= SET * /RST
AREG.RSTF=RST * /SET
```

這個意思是說 SET=1 RST=0 將使暫存器輸出腳 AREG 設定為 1

SET=0 RST=1 將使暫存器輸出腳 AREG 設定為 0

## (21) START\_UP (一開機即可跳入指定的狀態)

```
START_UP := POWER_UP ->STATE NAME
          +CONDITION1 -> STATE NAME
```

一開機若輸入為 POWER\_UP 則跳至 STATE NAME 狀態

一開機若輸入為 CONDITION1 則跳至 STATE NAME 狀態

## (22) SETF (輸入值設定)

```
SETF  A /B /C
```

設定輸入信號 A=1

B=0

C=0



## (23) STRING (用關鍵字代替方程式或一連串的字元)

例如：

```
STRING X1 'B1+/B2+B3 '
```

```
STRING X2 'B1+B2+B3 '
```

EQUATIONS

```
BANK1 = X1
```

```
C1 = /X1
```

```
C2 = X2
```

當組譯時 X1 即被  $B1+/B2+B3$  所代替

當組譯時 X2 即被  $B1+B2+B3$  所代替

## (24) .TRST (可單獨控制暫存器輸出之三態閘是否致能)

例如：圖 B-2

A. TRUST=VCC (A 暫存器輸出致能)

B. TRUST=GND (B 暫存器輸出除能)

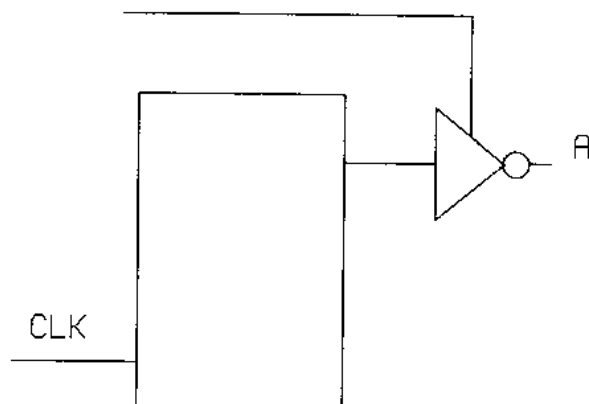


圖 B-2 輸出具三態閘

## (25) TRACE\_ON (指定欲追蹤的信號及顯示的順序)

```
TRACE ON A' B' C' D
```

當組譯完畢產生的模擬檔 .TRF 記錄這些追蹤信號。

在主功能表 VIEW 欄位選擇 WAVEFORM 欄位 再選擇 TRACE 螢幕出現 .TRF 之圖形，描述 A B C D 之內容。

## (26) TRACE\_OFF 結束 TRACE\_ON 之命令。

## (27) WHILE ...DO (當某一條件符合重覆執行一連串的指令)

```
TRACE_ON SET /OE CLOCK /OE A B C
SETF /SET OE /CLOCK
WHILE ( A * /B * C ) DO
  BEGIN
  CLOCKF CLOCK
  END
TRACE_OFF
```

當  $A * /B * C = 1$  則重覆執行 CLCOKF CLOCK 指令。

## (28) OPERATOR (各種運算子)

| 運算子 | 定義           | 例如            |
|-----|--------------|---------------|
| *   | NOT          | /X            |
| +   | OR           | C+D           |
| :+: | XOR          | C:+:D         |
| ::: | XNOR         | C>:::D        |
| =   | 組合方程式        | A = B + C     |
| :=  | 暫存器 EQUATION | A := B * C    |
| :=  | 轉移方程式        | S1:= S0 ->END |
| ->  | 狀態轉移         | S1:= S0 ->END |

## PLD數位邏輯實務應用



本書以簡單實例使讀者輕易瞭解 PAL 的基本構造，並說明如何使用 PALASM 4 軟體，對每一例題先以數位邏輯的觀點分析，並選擇適當的 PAL，再以 PALASM 4 的語法描述之，因此熟讀本書將使讀者更進一步瞭解數位邏輯，並快速的學會 PALASM 4 程式的撰寫。



### 磁片簡介

本學習磁片含有書中所列組合電路例題11個PDS檔及循序電路例題10個PDS檔。



### 作者簡介

姓名：陳兆麟  
現職：亞東工專電子工程科教師  
學歷：交通大學資訊科學研究所  
著作：PCAD入門及應用實例  
PC/AT 286、386線路  
分析及檢修實務

 碁亞資訊股份有限公司  
GOTOP INFORMATION INC.

台北市南港路三段50巷7號5樓  
TEL:(02)788-2408 FAX:(02)788-1031  
香港葵涌興芳路223號新都會廣場二期1905室  
TEL:(852)2484-9423 FAX:(852)2484-9269  
新竹市光復路二段882號5樓之6  
TEL:(03)5611-714 FAX:(03)5611-704  
台中市河南路二段262號8樓之7  
TEL:(04)255-8051 FAX:(04)255-8053  
高雄市三民區水源路18號7樓之1  
TEL:(07)389-6666 FAX:(07)385-3987

### 使用者指引

- |                                        |                                          |
|----------------------------------------|------------------------------------------|
| <input checked="" type="checkbox"/> 入門 | <input checked="" type="checkbox"/> 教材   |
| <input checked="" type="checkbox"/> 進階 | <input type="checkbox"/> 參考手冊            |
| <input type="checkbox"/> 高階            | <input checked="" type="checkbox"/> 使用指南 |
|                                        | <input type="checkbox"/> 應用技巧            |



9789576415715  
02849111 NT\$ 300