

cab 條碼機程式手冊

簡介

cab 條碼機可使用的三種型式指令：

- **ESC** 指令
- 小寫英文字母的指令
- 大寫英文字母的指令

ESC 指令

用來檢測條碼機狀態、控制條碼機、與記憶體管理，且 ESC 指令通常是會立刻執行，即使是條碼機正在列印標籤的狀態；ESC 指令並非都用在列印標籤，還有其他功能，例如：

ESC ? = 查詢可用的記憶體狀態

ESC c = 取消列印

ESC p0 = 結束條碼機暫停狀態

小寫英文字母的指令

用來調整及設定條碼機，與實際列印無關連，例如：

a = 條碼機啟動 ASCII 碼列印模式

c = 條碼機執行立刻裁切

f = 條碼機執行進紙功能

大寫英文字母的指令

用來編寫標籤格式，並依據固定結構與語法，例如：

J = 標籤設計開始

S = 設定標籤尺寸

H = 設定印字頭工作溫度、列印速度與列印方式

O = 設定列印方向

T = 標籤文字定義

B = 標籤條碼定義

G = 標籤線條定義

I = 標籤圖形定義

A = 標籤列印數量

另外，cab 條碼機還支援特殊功能指令，如 **[DATE]** 列印日期指令，與資料庫連結指令，細節請參考後續解說。此程式手冊所提供的範例是以公制毫米（mm）為單位，以 A3/300(dpi)印表機列印結果為圖示，而標籤編寫都可以 Notepad、Wordpad 等文書編輯軟體編寫

ESC 指令 - ESCESC

該指令用在當下載圖形或字型到條碼機時，圖形或字型可能會內含相同 ESC 指令，故以 ESCESC 替代單一 ESC（ASCII 27 或 Hex 1B 碼）指令以避免無預期的錯誤反應；

語法： **ESCESC**

ESC 指令 - ESC ! ESC !

該指令會強迫條碼機重開機，功能如同手動開關條碼機電源；

語法： **ESC ! ESC !**

ESC 指令 - ESC*

該指令會透過 RS-485 連線啟動所有連結的條碼機，此時必須要所有的條碼機皆有正確設定 RS-485 連線網路 ID 才可，最多可接 26 台條碼機，RS-485 的網路位址代號為 A-Z；

語法： **ESC***

語法： **ESCA**

:

語法： **ESCZ**

ESC 指令 - ESC.

在傳輸圖形或字型等二進制資料時，該指令可以二進位資料起始與結束值，此功能亦可透過 “cab cardmanager” 軟體做資料傳輸的功能；

語法： **ESC.**

ESC 指令 - ESC :

該指令用在二進制資料的敘述，通常在下載資料到條碼機時，需要先將資料轉換，因此便需要起始順序，再接二進制資料，最後再接 ESCend-of-data，而二進制資料不能包含任何 ESC 字元，否則會被系統錯誤解譯；該指令不能用於網路，下載二進制資料最好方式是使用 ESC. 指令；

語法： **ESC :**

ESC 指令 - ESC?

該指令會查詢條碼機記憶體緩衝區的使用情形，會以 0-9 數值回傳狀態；

語法： **ESC?**

數值		記憶體使用率
0	=	0-9%
1	=	10-19%
2	=	20-29%
3	=	30-39%
4	=	40-49%
5	=	50-59%
6	=	60-69%
7	=	70-79%
8	=	80-89%
9	=	90-99%

ESC 指令 - ESCc

該指令會取消條碼機當時的列印動作，相當於按下條碼機 LCD 控制面版上的 CANCEL 鍵；

語法： **ESCc**

ESC 指令 - ESCend-of-data

該指令用在二進制資料的敘述的結束，搭配 **ESC**：指令使用，但不能用於 RS-485 網路；

語法： **ESCend-of-data**

ESC 指令 - ESCf

該指令會執行條碼機的進紙動作，相當於按下條碼機 LCD 控制面版上的 FEED 鍵；

語法： **ESCf**

ESC 指令 - ESCp0

該指令會終止條碼機的暫停模式，此時條碼機 LCD 控制面版上的 PAUSE LED 指示燈會熄滅，並繼續未完成的列印動作，或是結束條碼機錯誤狀態，如同按下 LCD 控制面版上的 PAUSE 鍵；

語法： **ESCp0**

ESC 指令 - ESCp1

該指令會使條碼機立刻進入暫停模式，如同按下 LCD 控制面版上的 PAUSE 鍵，此時條碼機 LCD 控制面版上的 PAUSE LED 指示燈會亮起，並暫停列印動作；

語法： **ESCp1**

ESC 指令 - ESCt

該指令會使條碼機完全取消所有列印，並清除條碼機緩衝區的資料，如同按下 LCD 控制面版上的 CANCEL 鍵，與 **ESCc** 指令不同處在於 **ESCc** 僅取消當時的列印動作，並未清除條碼機緩衝區的資料；

語法： **ESCt**

ESC 指令 - ESCs

該指令會透過 RS-232 條碼機連結，查詢條碼機狀態並回傳狀態訊息，

語法：**ESCs**

回傳訊息：**XYNNNNNNNZ**

說明：**X** = 條碼機的連線狀態，

Y = 條碼機與電腦為連線狀態，**N** = 未連線

Y = 條碼機錯誤代號

NNNNNN = 要列印的標籤數量

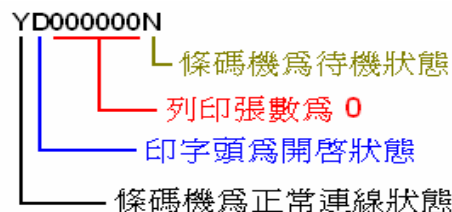
Z = 條碼機列印狀態，**Y** = 正在處理列印，**N** = 條碼機處於待機模式

其中 **Y** 的代號意義如下：

代號	說明	解決方式
-	無錯誤	
B	通訊協議錯誤	確定電腦與條碼機的通訊協議設定皆為相同。
C	記憶卡錯誤	重新格式化記憶卡，再存入標籤檔案。
D	印字頭為開啓狀態	關閉印字頭。
E	同步化錯誤（無標籤）	檢查標籤紙與標籤感應器是否對準標籤與標籤之間間距位置。
F	碳帶用完	更換碳帶。
H	印字頭加熱電壓問題	關閉電源再重新開機，若還有相同問題，請聯絡專業人員服務。
O	記憶體不足	清除記憶體或重新開機。
P	標籤紙用完	更換標籤紙捲。
V	輸入緩衝溢位	重新開機。
W	印字頭過熱	暫停列印一段時間。
X	外部 輸入/輸出 錯誤	檢查外接裝置。
Z	印字頭損毀	清潔或更換印字頭。
n	網路錯誤	檢查網路設定與硬體安裝。
u	USB錯誤	檢查USB裝置與連接。

範例：

YD000000N



條碼機為待機狀態

列印張數為 0

印字頭為開啓狀態

條碼機為正常連線狀態

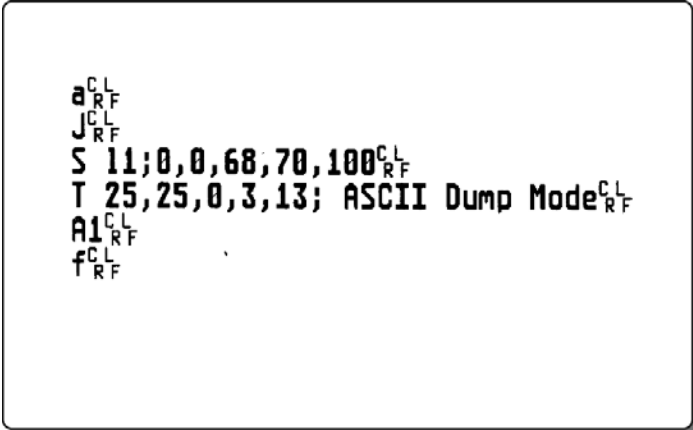
小寫英文字母的指令

a - ASCII 碼列印模式

語法： **a**

範例：

```
a
J
S 11,0,0,68,70,100
T 25,25,0,3,13;ASCII Dump Mode
A1
f
```



```
a(97)
J(4A)
S 11(0B),0(00),0(00),68(44),70(46),100(64)
T 25(19),25(19),0(00),3(03),13(0D); ASCII Dump Mode
A1(41)
f(66)
```

C - 立刻裁切

語法： **c**

c = 條碼機立刻執行裁切動作，如條碼機未裝裁刀，則會在 LCD 上顯示 “Protocol error” 的錯誤訊息；

e - 清除資料

語法： **e type;name**

- e** = 清除條碼機記憶體資料（如圖形或字型等），但此指令並不會清除條碼機的記憶卡資料；
- type=** 清除檔案的格式，如 BMP、FNT、GIF、IMG、MAC、PCX、PNG、TIF、TTF 等檔案
- name=** 檔案的名稱，如使用 * 作為檔名，則會刪除所有相同格式的檔案！

範例：

E FNT;*

說明：刪除條碼機記憶體內所有的 True Type 字型；

f - 條碼機進紙

語法： **f**

f = 執行條碼機進紙動作的指令，如同按下控制面版上的 FEED 鍵；

m - 設定長度單位

語法： **m t**

- m** = 設定長度單位指令
- t** = 單位選擇，“m”是公制(毫米，mm)，“i”是英制(英吋，inches)

p - 暫停條碼機

語法： **pn**

- p** = 條碼機暫停指令
- n** = 0：取消暫停
1：啟動暫停

啟動條碼機暫停時，相當於按下條碼機的 PAUSE 鍵，此時 PAUSE 的 LED 指示燈會亮起，如當時有列印標籤，則會立刻暫停列印動作；當取消暫停時，PAUSE 的 LED 指示燈會熄滅，如當時有被暫停列印標籤，則會恢復列印動作。

大寫英文字母的指令

A – 標籤數量

A 指令是用來定義標籤列印數量，在條碼機執行列印時，標籤設計會儲存在條碼機的內部記憶體緩衝區直到完成列印動作；

語法： **A n**

- n** = 標籤數量，如未輸入數目，則會無限列印；
- 其他有效參數
- [NOPRINT]** = 條碼機回接收並處理標籤資料，但無列印動作，此目的是將標籤資料儲存於條碼機記憶體，亦可使用 [NO] 代替 [NOPRINT] 指令；
- [?]** = 條碼機會在LCD面版上顯示要求輸入列印數量；
- [REPEAT]** = 在列印工作結束時，重複循環標籤資料，此功能常與 [?] 結合使用，亦可以 [R] 代替 [REPEAT] 指令
- [\$DBF]** = 列印資料庫的每筆記錄，資料庫的總記錄數目就是標籤列印張數

範例：

```
J
S 11;0,0,68,71,100
T 25,10,0,5,8;LABEL PRINTER
A 550
說明：列印 550 張標籤
```

範例：

```
J
S 11;0,0,68,71,100
T 25,10,0,5,8;LABEL PRINTER
A
```

說明：

A 指令後面不加任何指令，則會列印無限張數的標籤，直到使用者輸入取消訊號或按下 LCD 控制面版上的取消鍵（Cancel）為止

範例：

```
J
S 11;0,0,68,71,100
T 25,25,0,3,8;Suppress Printout
A [NOPRINT]
```

說明：

利用 [NOPRINT] 指令將標籤資料傳送到條碼機記憶體緩衝區

範例：

```
J
S 11;0,0,68,71,100
T 25,25,0,3,8;[?:Input?]
A [?,R]
```

說明：

此標籤程式會在條碼機上的 LCD 控制面版上顯示要求操作者輸入標籤資料 ([?:Input?])，然後會再顯示要求操作者輸入標籤列印張數，完成輸入後才會開始列印標籤，所有資料輸入可由 LCD 控制面版的按鍵輸入，或由外接鍵盤輸入。

範例：

```
m m
J
S 11;0,0,68,73,100
E DBF;CDPLAYER
T:IDX;25,225,0,3,5;[SER:100]
T0,40,0,3,6;>>[DBF:TYP,typ,NAME]<<
A [$DBF]
```

說明：

列印資料庫 CDPLAYER.DBF 裡所有的紀錄，序列號指令 ([SER:100]) 會建立索引檔，從 100 開始起跳。

B - 條碼

條碼定義

各條碼參數不盡相同，依條碼類型而定，條碼方向可為 0°、90°、180°、270°，高度與寬度皆可調整，亦可加入文字以利人工判讀。

語法：**B [:name;] x,y,r,type[+options],size;text**

B	=	定義條碼指令
[:name;]	=	定義資料內容於唯一的資料名
x	=	X 軸位置
y	=	Y 軸位置
r	=	旋轉角度，0~359 度間隔 1°
type	=	條碼類型
[+options]	=	附加參數
size	=	條碼高度、寬度與比例
text	=	條碼內容

解釋如下

B

條碼指令，指令後接參數設定與條碼內容；

[:name;]

定義資料內容於唯一的資料名，供後續程式呼叫或其他用途(如數學運算)使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；

x

條碼水平 X 軸起始位置，單位為 mm 或英吋；

y

條碼垂直 Y 軸起始位置，單位為 mm 或英吋，最大座標會依條碼機類型而定；

r

旋轉角度，可為 0°、90°、180°、270°；

type

條碼類型，如以英文字母大寫定義條碼類型，則條碼會附帶人工判讀字元，如以小寫字母定義條碼類型，則不會有人工判讀字元的產生，人工判讀字元的大小會依不同條碼類型而有所不同，細節部分可參考後面範例；

cab 條碼機能從條碼名稱裡提取重點部分以判定條碼類型，如指令行裡為 EAN-13、EAN 13 或 EAN13 通通都會判定為 EAN 13 碼；

[+options]

依據條碼類型可有不同附加功能，不同條碼有的不同附加功能會在各條碼解說裡詳細敘述，有效的附加功能如下：

+MODxx

可在條碼裡加入檢查碼

MOD10	加入檢查碼 10
MOD11	加入檢查碼 11
MOD43	加入檢查碼 43
MOD16	加入檢查碼 16

+WSarea

白色空白區域-可列印（保留）白色區域，以便讓列印之條碼有更好的掃描判讀性；

**+BARS**

在條碼上下方列印直線的邊線；

+XHRI

39 碼：（附加人工可讀的內容）添加起始與結束字元（*）；

93 碼：添加起始與結束字元（□）；

減小 UPC-A 與 UPC-E 的大小

+NOCHECK

隱藏專為 EAN-13 與 UPC-A 碼的特別起始數字 21、24...29 設計的變量重量條碼的檢查計算碼；

+ELx

錯誤層級，設定 PDF417 碼的冗位，x 的有效值為 0~8；

DtatMatrix 碼可列印成矩形或正方形，預設值為正方形，可用+RECT 指令列印成矩形。

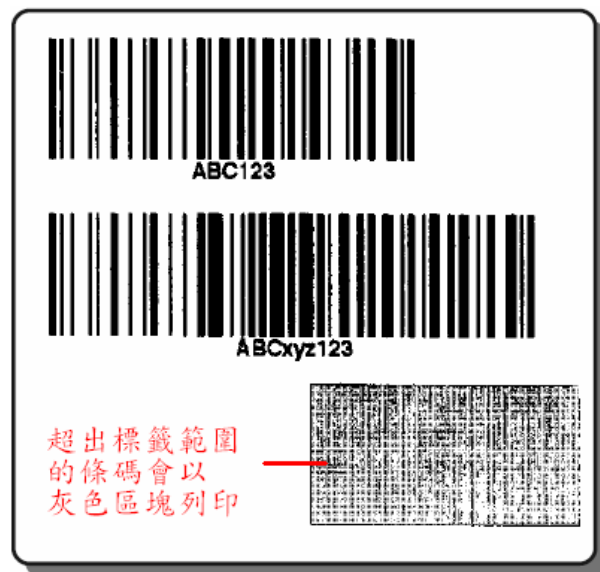
size

定義條碼高度與寬度，EAN、JAN 及 UPC 碼亦可以 SCx 定義標準碼大小，高度計算在開啓人爲判讀的字元時，包含該字元高度；

height - 高度

以預設單位-毫米(mm)或英吋(inches)定義條碼高度，如條碼大小（含

空白區域) 超出標籤紙時，A 系列條碼機會將該條碼列印成灰色區塊，如下圖所示：



narrow element (ne) - 窄度

以毫米或英吋單位定義條碼最小窄度，窄度大小會依條碼機解析度而不同；

ratio - 比例

比例為條碼窄度與寬的比，如 3：1 表示寬的碼條為小碼條的三倍寬；

SCx

SC 為標準條碼尺寸，EAN 與 UPC 碼為統一尺寸；

text

依所選擇的條碼類型所包含條碼的資料，不同條碼有不同條碼資料定義，例如，部分條碼僅能使用數字，部分條碼則有固定資料長度；

A 系列條碼機會將超出標籤紙的條碼印呈灰色區塊，可方便人為判別該條碼為有問題的條碼，以便免後續的讀碼問題。

B - 條碼

條碼總覽表

不同條碼會有不同的尺寸比例，條碼名稱爲大寫表示會有人判讀的文字行，此部分可參考該條碼的規範；

Barcode name	Ratio	1D /2D code*	A-series	Apollo	Hermes	M-series
2 of 5 Interleaved	yes	1D	yes	yes	yes	yes
Add-On 2	no	1D	yes	yes	yes	yes
Add-On 5	no	1D	yes	yes	yes	yes
Aztec Code	no	2D	yes	-	-	yes
Codabar	yes	1D	yes	yes	yes	yes
Codablock F	no	stacked	yes	-	-	yes
Code 39	yes	1D	yes	yes	yes	yes
Code 93	no	1D	yes	yes	yes	yes
Code 128	no	1D	yes	yes	yes	yes
Data Matrix	no	2D	yes	yes	yes	yes
DBP (German Post code)	yes	1D	yes	yes	yes	yes
EAN 8	no	1D	yes	yes	yes	yes
EAN 13	no	1D	yes	yes	yes	yes
EAN 128	no	1D	yes	yes	yes	yes
FIM	no	1D	yes	yes	yes	yes
German Parcel	yes	1D	yes	yes	yes	yes
JAN 8	no	1D	yes	yes	yes	yes
JAN 13	no	1D	yes	yes	yes	yes
HIBC	yes	1D	yes	yes	yes	yes
MaxiCode	no	2D	yes	yes	yes	yes
Micro PDF	no	2D	yes	-	-	yes
MSI	yes	1D	yes	yes	yes	yes
PDF-417	no	2D	yes	yes	yes	yes
Plessey	yes	1D	yes	yes	yes	yes
Postnet	no	1D	yes	yes	yes	yes
QR -Code	no	2D	yes	-	-	yes

1D 表示一維條碼，2D 表示二維條碼

Barcodename	Ratio	1D /2D code*	A-series	Apollo	Hermes	M-series
RSS-14		1D	yes	-	-	yes
RSS-14 composite CC-A		1D+2D	yes	-	-	yes
RSS-14 composite CC-B		1D+2D	yes	-	-	yes
RSS-14 truncated		1D	yes	-	-	yes
RSS-14 truncated composite			yes	-	-	yes
RSS-14 truncated composite			yes	-	-	yes
RSS-14 stacked			yes	-	-	yes
RSS-14 stacked composite			yes	-	-	yes
RSS-14 stacked composite			yes	-	-	yes
RSS-14 stacked omnidirectional			yes	-	-	yes
RSS-14 stacked omnidirectional composite			yes	-	-	yes
RSS-14 stacked omnidirectional composite			yes	-	-	yes
RSS limited			yes	-	-	yes
RSS limited composite			yes	-	-	yes
RSS limited composite			yes	-	-	yes
RSS expanded			yes	-	-	yes
RSS expanded composite			yes	-	-	yes
RSS expanded composite			yes	-	-	yes
RSS expanded stacked			yes	-	-	yes
RSS expanded stacked half line			yes	-	-	yes
RSS expanded stacked composite (CC-A)			yes	-	-	yes
RSS expanded stacked composite (CC-B)			yes	-	-	yes
UCC 128	no	1D	yes	yes	yes	yes
UPC-E0	no	1D	yes	-	-	yes
UPC-A	no	1D	yes	yes	yes	yes
UPC-E	no	1D	yes	yes	yes	yes

此程式手冊不包含 Aztek、Codablock、與 RSS 碼！

B - 條碼

每個條碼都有個別條碼資料與型式的定義，建議使用者先瞭解個別條碼的定義規範，同時也建議先測試列印出來條碼的可讀性！

有效的檢查碼：

MOD 10（僅數字資料）

MOD 10（MSI 碼用以計算差值（weighting 為 2/1 而非 3/1）

MOD 10 GP（2 of 5，weighting 3/1 + 1，僅用在德國包裹）

MOD 11（僅數字資料）

MOD 16（僅 Codabar）

MOD 43（僅 39 碼與 128 碼）

Code 128 與 EAN/UCC-128 碼會自動使用 103 檢查碼；

EAN-13、EAN-8、UPC-A、與 UPC-E0 會自動使用檢查碼 10；

POSTNET 會自動使用檢查碼 10（無 weighting）；

DBP 為 Deutsche Post AG 的 12 或 14 碼條碼，會自動使用檢查碼 10（weighting 4/9），可依要求加入空白與點。

B - 條碼 2 of 5 Interleaved

語法： **B** [:name;] x,y,r,**2OF5INTERLEAVED**[+options],height,ne,ratio;text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+MODxx

可加入檢查碼到條碼裡；

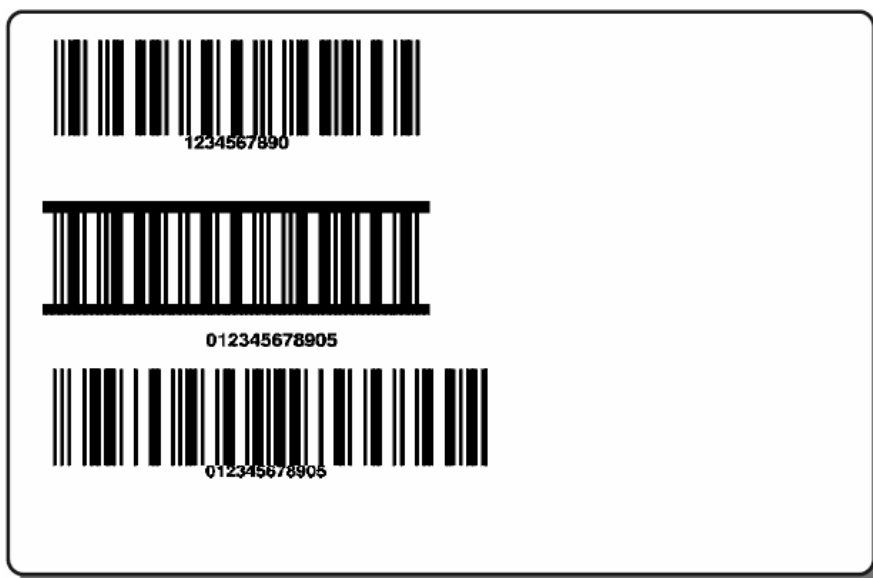
+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

建議該條碼使用固定長度，並設定條碼掃描器為相同的條碼長度，以確保最佳的條碼判讀性。

範例：

```
J
S 11;0,0,68,71,100
B 5,5,0,2 OF 5 INTERLEAVED,10,.3,3;1234567890
B 5,20,0,2of5interleaved+BARS,10,.3,3;1234567890
B:Bar3;5,35,0,2OF5 INTERLEAVED+MOD10,10,.3,3;1234567890
A 1
```



B - 條碼 Add-On2

語法： **B** [:name;] x,y,r,ADDON2,[+options],height,ne;text

[+options]=

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10,5,0,EAN13 ,SC2;402345607891
B 45,5,0,ADDON2,SC2;09
A 1
```



B - 條碼 Add-On5

語法： **B** [:name;] x,y,r,ADDON2,[+options],height,ne;text

[+options]=

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10,5,0,EAN13, SC2;402345607891
B 45,5,0,ADDON5,SC2;00399
A 1
```



B - 條碼 Codabar

語法： **B [:name;] x,y,r,CODABAR[+options],height,ne,ratio;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+MODxx

可加入檢查碼到條碼裡；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,CODABAR,      12,.3,3;A12345678A
B 5,20,0,CODABAR, 12,.3,3;A23456789C
B 5,35,0,CODABAR+MOD16,12,.3,3;A13572468C
A 1
```



B - 條碼 Code 39

語法： **B [:name;] x,y,r,CODABAR[+options],height,ne,ratio;text**

[+options]=

+WSarea

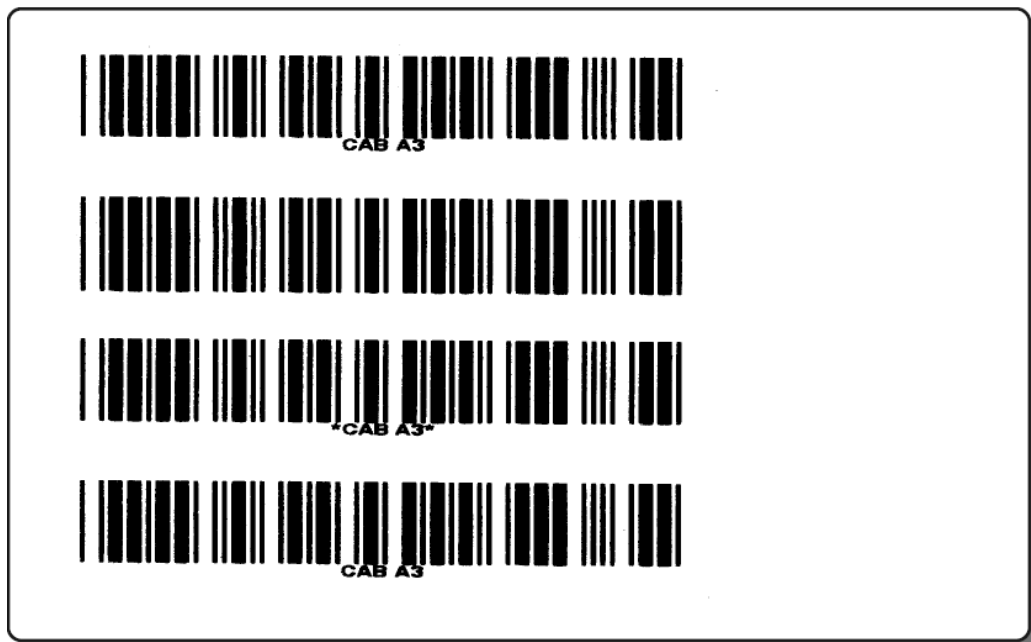
在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+XHRI

人工判讀內容，並加入起始與結束字元；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,CODE39,10,0.3,3;CAB A3
B 5,20,0,code39,10,.3,3;CAB A3
B 5,35,0,CODE39+XHRI,10,0.3,3;CAB A3
B 5,50,0,CODE39,10,.3,3;cab A3
A 1
```



B - 條碼 Code 93

語法： **B [:name;] x,y,r,CODE93,[+options],height,ne;text**

[+options]=

+BARS

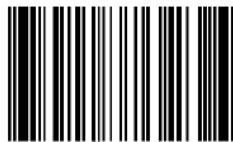
在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

+XHRI

人工判讀內容，並加入起始與結束字元 (□)；

範例：

```
J
S 11;0,0,68,71,100
B 5,5,0,CODE93+XHRI,16,.28,3;ABC123
B 5,24,0,code93,16,.28,3;ABC123
B 5,44,0,CODE93+BARS,16,.28,3;ABC123
A 1
```



□ABC123□



ABC123

B - 條碼 Code 128

語法： **B** [:name;] x,y,r,**CODE128**[+options],height,ne,ratio;[U:subcode]text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+MODxx

可加入檢查碼到條碼裡；

+BARS

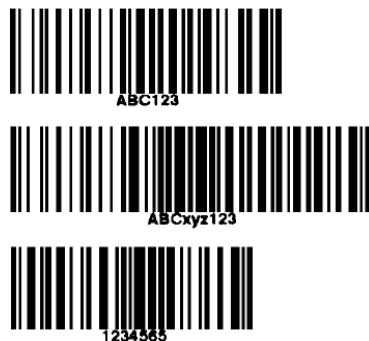
在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

[U:subcode]

可選用特定的條碼子編碼；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,CODE128, 12,.3;ABC123
B 5,20,0,CODE 128,12,.3;ABCxyz123
B 5,35,0,CODE128+MOD10, 12,.3;[U:CODEC] 123456
A 1
```

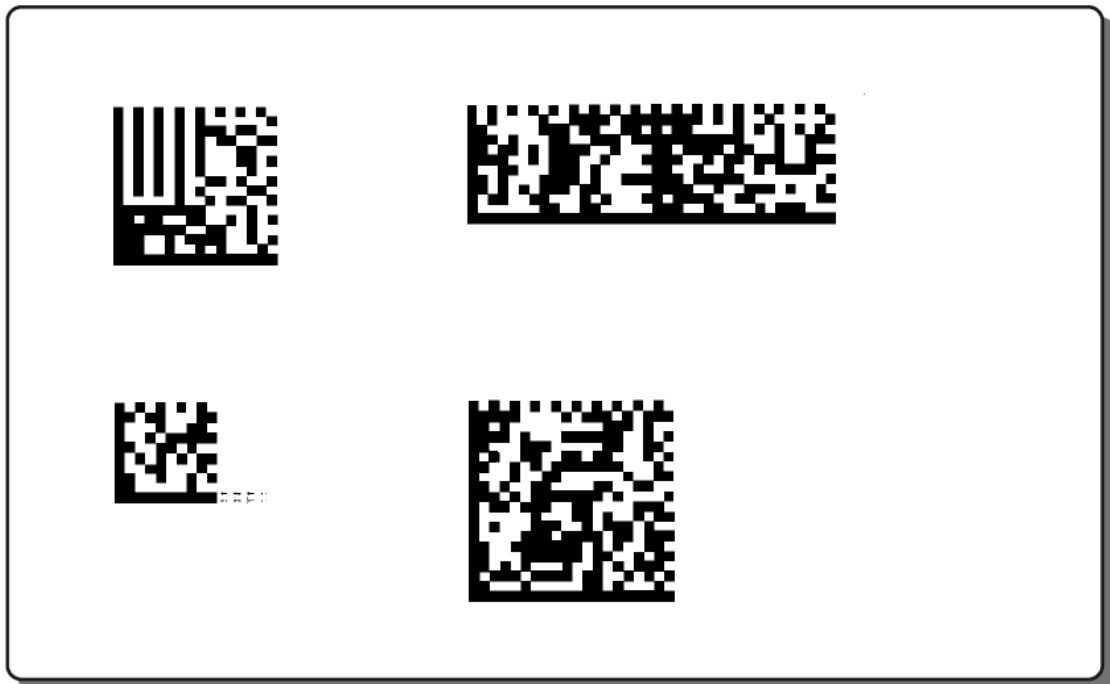


B - 條碼 Data Matrix

語法： **B** [:name;] x,y,r,DATAMATRIX [+RECT],height;text

範例：

```
J
S 11;0,0,68,71,100
B 25, 5,0,DATAMATRIX,1;30Q324343430794<OQQ
B 60, 5,0,DATAMATRIX+RECT+WS2,1;cab Produkttechnik
B 25,35,0,DATAMATRIX,1;[U:PROG]
B 60,35,0,DATAMATRIX+WS2,1;[U:ANSI_AI] cabProdukttechnik
A 1
```



B - 條碼 DBP-German Post Identcode

語法： **B** [:name;] x,y,r,DBP[+options],height,ne,ratio;text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

範例：

```
J
S 11;0,0,68,71,100
B 5,10,0,DBP,10,.3;2134807501640
B 60,10,0,DBP,10,.3;56.310.243.031
A 1
```



21348.075.016.40 1



56.310.243.031 3

B - 條碼 EAN-8 / JAN-8

語法： **B** [:name;] x,y,r,EAN8[+options],height,ne;text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+XHRI

人工判讀內容，並縮小條碼大小（參見範例）；
高度與窄度可以 SC 參數設定；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10, 5,0,EAN8, SC1;4023456
B 10,26,0,EAN8,16,.35;4023456
B 10,44,0,JAN8,16,.35;4900056
A 1
```



B - 條碼 EAN-13 / JAN-13

語法： **B [:name;] x,y,r,EAN13[+options],height,ne;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+XHRI

人工判讀內容，並縮小條碼大小（參見範例）；

+NOCHECK

隱藏 EAN-13 碼的特別起始數字：21、24...29 之變量重量條碼的檢查計算碼；

高度與窄度可以 SC 參數設定；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10, 5,0,EAN13, SC1;402345607891
B 10,30,0,EAN13,16,.35;270072610950
B 10,48,0,JAN13,16,.35;490005607891
A 1
```



B - 條碼 EAN-128 / UCC 128

語法： **B** [:name;] x,y,r,EAN128[+options],height,ne;text

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,EAN128,12,.3;(00)345678901234567890
B 5,20,0,UCC128,12,.3;(00)345678901234567890
B 5,35,0,EAN128,      12,.3;(00)345678901234567890
A 1
```



B - 條碼 FIM

語法： **B [:name;] x,y,r,FIM,[+options],height,ne;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,FIM,16,.3,3;A
B 5,24,0,FIM,16,.3,3;B
B 5,44,0,FIM, 16,.3,3;C
A 1
```



B - 條碼 HIBC (Health Industry Barcode)

語法： **B [:name;] x,y,r,HIBC[+options],height,width,ratio;text**

[+options]=

+WSarea

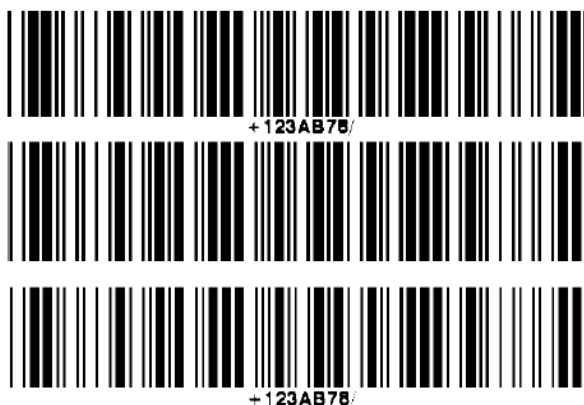
在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,HIBC,12,.3,3;+123AB78
B 5,18,0,hibc,12,.3,3;+123AB78
B 5,33,0,HIBC, 12,.3,3;+123AB78
A 1
```



B - 條碼 Maxicode

語法： B [:name;] x,y,r,MAXICODE[+MODE];[ZIPCODE],[COUNTRY]...

範例：

```
J
S 11;0,0,68,71,100
B 25,5,0,Maxicode+MODE2;76131,260,999,Paket for cab
Produkttechnik GmbH
B 60, 5,0,Maxicode+ws2+mode4;MaxiCode (19 charcters)
B 25,35,0,Maxicode+MODE4;Paket for cab Produkttechnik GmbH
B 60,35,0,Maxicode+MODE6;Paket for cab Produkttechnik GmbH
A 1
```

B - 條碼 Micro PDF 417

語法： **B [:name;] x,y,r,Micro[+COLSx],height,ne,ratio;text**

COLSx = 設定 PDF 碼的資料行數，x=1-4

Micro PDF417 提供三種編碼方式：

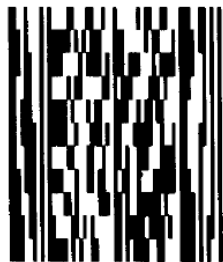
文字壓縮模式 **mode0**：250 個字元，每 2 個資料字元壓縮成一個碼

位元壓縮模式 **mode1**：150 個字元，每 1.2 個資料字元壓縮成一個碼

數值壓縮模式 **mode2**：366 個字元，每 2.93 個資料字元壓縮成一個碼

範例：

```
J
S 0,0,68,71,100
B 10,10,0,Micro+COLS2,3,.5;cab Produkttechnik
A 1
```



B - 條碼 MSI (MSI Plessey)

語法： **B [:name;] x,y,r,MSI[+options],height,ne,ratio;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+MODxx

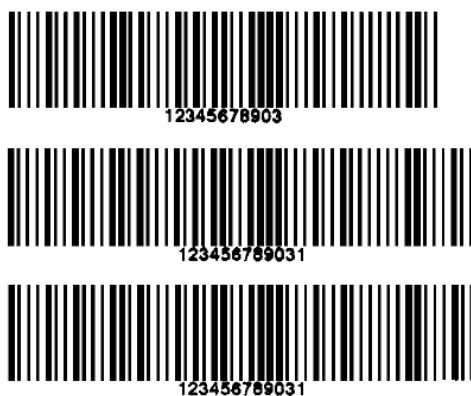
可加入檢查碼到條碼裡；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

範例：

```
J
S 11;0,0,68,71,100
B 5, 5,0,MSI,12, .3,2;1234567890
B 5,20,0,MSI+MOD10,12, .3,2;1234567890
B 5,35,0,MSI+MOD11,12, .3,2;1234567890
A 1
```



B - 條碼 PDF417

語法： **B [:name;] x,y,r,PDF417[+WSarea],[+ELxx],height,ne,ratio;text**

[+options]=

+WSarea

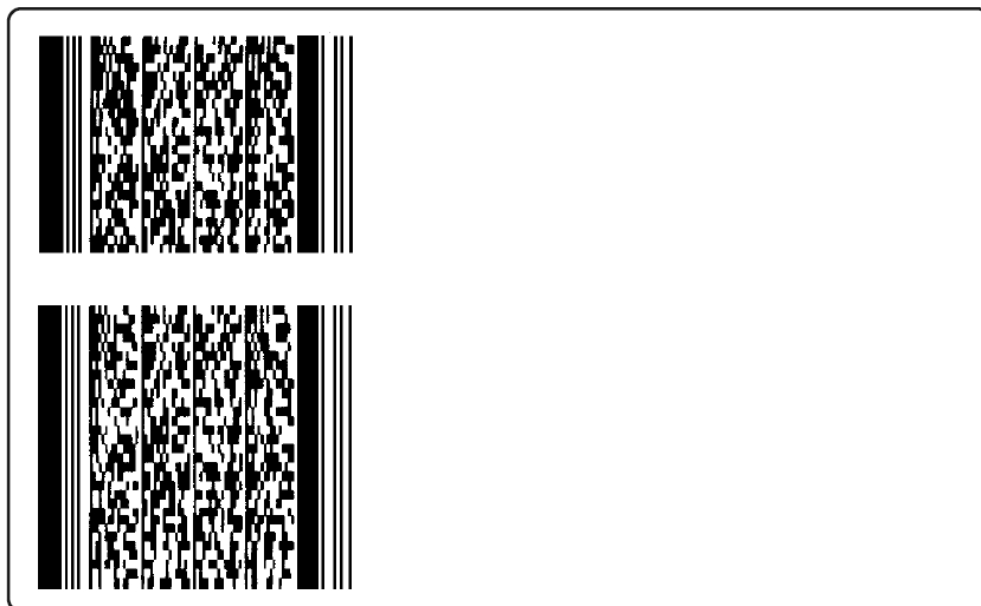
在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+ELxx

錯誤層級由此數值設定；

範例：

```
J
S 11;0,0,68,71,100
B 2, 5,0,PDF417+EL0,.1,.38,1;cab Produkttechnik
GmbH[U:13][U:10]Wilhelm Schickard Strasse[U:13][U:10]D-76131
Karlsruhe
B 2,35,0,PDF417+EL3,.1,.38,1;cab Produkttechnik
GmbH[U:13][U:10]Wilhelm Schickard Strasse [U:13][U:10]D-76131
Karlsruhe
A 1
```



B - 條碼 Plessey

語法： **B** [:name;] x,y,r,PDF417[+WSarea],[+ELxx],height,ne,ratio;text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

範例：

```
J
S 11;0,0,68,71,100
B 5,20,0,PLESSEY+BARS,12,.3,2;1234567890
B 5,35,0,plessey,12,.3,2;1234567890
A 1
```



B - 條碼 Postnet

語法： **B** [:name;] x,y,r,POSTNET,[+options];text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

範例：

```
J
S 11;0,0,68,71,100
B 10, 5,0,postnet,20,.35;442120798
B 10,20,0,POSTNET,          20,.35;441361234
A 1
```



B - 條碼 QR-Code

語法： **B [:name;] x,y,r,QRCODE[+ELx][+MODELx],size;text**

+ELx

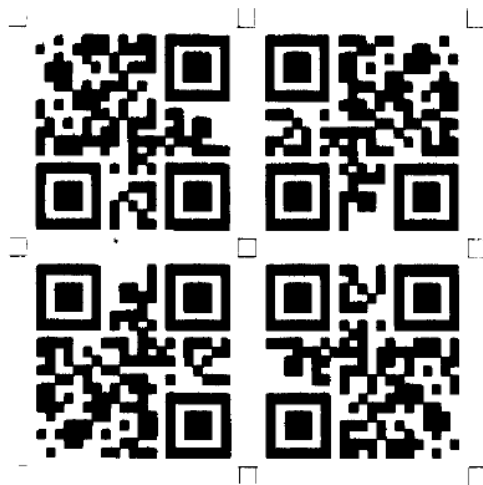
錯誤層級 – 有效數值 x：1-4, L, M, Q, H，預設值為 1

+MODELx

有效輸入值 x 為 1 或 2，預設值為 1

範例：

```
J
S 11;0,0,68,71,104
B 52,32,0,QRCODE+ELL+MODEL2+WS2,1;Hello world!
B 52,28,90,QRCODE+ELL+MODEL2+WS2,1;Hello world!
B 48,28,180,QRCODE+ELL+MODEL2+WS2,1;Hello world!
B 48,32,270,QRCODE+ELL+MODEL2+WS2,1;Hello world!
G 0,0,0;L:104,3
G 0,65,0;L:104,3
H 150,-5,T
A 5
```



B - 條碼 **UPC-A**

語法： **B** [:name;] x,y,r,UPCA[+options],height,ne;text

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+XHRI

人工判讀內容，並縮小條碼大小（參見範例）；

+NOCHECK

隱藏 UPC-A 碼的特別起始數字：21、24...29 之變量重量條碼的檢查計算碼；

高度與窄度可以 SC 參數設定；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
m m
J
O R
S 11;0,0,68,71,100
B 10,5,0,UPC-A,20,.35;01234554321
B 10,30,0,UPCA+XHRI,SC1;01234554321
A 1
```



B - 條碼 **UPC-E**

語法： **B [:name;] x,y,r,UPCE[+options],height,ne;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+XHRI

人工判讀內容，並縮小條碼大小（參見範例）；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10, 5,0,UPC-E,20,.35;0123456
B 10,30,0,UPCE+XHRI,SC1;0123456
A 1
```



B - 條碼 **UPC-E0**

語法： **B [:name;] x,y,r,UPCE0,height,ne;text**

[+options]=

+WSarea

在標籤周圍列印（保留）空白標記，以確保該條碼能被正確讀取，此功能僅用在設計目的，在標籤能正常列印與條碼能正確判讀後，應去掉此指令；

+BARS

在條碼上下方列印直線的邊線，讓條碼有更好的可讀性，也能避免該條碼的錯誤讀取；

SCx

設定條碼大小，x 為 0~9 數值，實際條碼大小會因條碼機解析度而不同；

範例：

```
J
S 11;0,0,68,71,100
B 10, 5,0,UPCE0,20,.35;03210000678
B 10,30,0,UPCE0,          SC1;01230000088
A 1
```



C - 裁刀參數

語法： **C amount [,disp1[.disp2]]**

C 啓動裁刀裁切指令
amount 每隔幾張標籤裁切一次
disp1 第一個裁切位移，以選定的長度單位作為依據
disp2 第一次裁切後的距離 disp2 再裁切一次，
 此參數必為正數值！

長度單位是以毫米(mm)或英吋(inch)為單位

語法： **C e**

C 啓動裁刀裁切指令
e 一直列印工作結束為止才裁切

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;cut after 2 labels
C2
A10
```

說明：

列印 10 張標籤，每兩張裁切一次。

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;cut after 2 labels
C5,0,2
A10
```

說明：

列印 10 張標籤，每 5 張裁切一次，裁切位移為 0，第一次裁切後的 2 mm 再裁切一次，也就是第六張標籤紙的前 2 mm 部分會被裁掉。

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;cut after 5 labels
C 5
A 100
R cut after 2 labels
C 2
A 60
```

說明：

先列印 100 張標籤時，每 5 張標籤裁切一次，然後再印 60 張標籤，每 2 張標籤裁切一次。

E - 檔案定義（延伸功能）

語法： **E EXT ; name_type**

E = 延伸檔案定義指令

EXT = 延伸檔案類型

有效的檔案類型：

DBF = Database 檔案，與 [DBF] 指令一起使用

TMP = 暫存檔，如跳序號檔案，
與 [RLOG] 及 [WLOG] 指令一起使用

LOG = 定義外部通訊協議檔案（LOG 檔）之名稱，
與 [RLOG] 及 [WLOG] 指令一起使用

SQL = 定義資料庫伺服器的位址，
與 cab Database Connector 軟體一起使用

name_type = 即 DBF、TMP、或 LOG 的檔名

= SQL 的 IP 位址與埠值（IP:port）

範例：

```
E DBF;article
```

說明：連結 CF 記憶卡裡的 article.DBF 檔，檔名必為 8.3 格式（8 個字元檔名與 3 個延伸字元副檔名）

範例：

```
E TMP;SERNUM
```

說明：連結 CF 記憶卡裡的 SERUM.TMP 序號暫存檔，檔名必為 8.3 格式

範例：

```
E LOG;PROTOCOL
```

說明：定義記憶卡內的 PROTOCOL.LOG 檔案，檔名必為 8.3 格式

範例：

```
E SQL;192.168.0.56:1001
```

說明：定義 SQL 的 IP 位址與埠值（1001）

F - 字型編號

語法： **F number;name**

F = 字型編號指令
number = 新的字型編號
name = 被字型編號取代的字型名稱

範例：

F 4;Times New Roman

說明：使用 TrueType 或 Speedo 名字型；

範例：

F 40; Swiss 721 Bold Italic

說明：定義條碼機內建字型 Swiss 721 Bold Italic 字型編號為 40；

範例：

```
J SAMPLE
H 66
S 11;0,0,68,71,100
F 10;Comix
T 0,15,0,10,pt20;SampleJ:c108]
T 10,25,0,3,pt12;label,
B 5,40,0,EAN-8,SC2;4376131
A 20
```

說明：定義下載到條碼機內部記憶體的字型（Comix）編號為 10，並列印兩行文字，第一行文字是以 Comix 字型列印，第二行則是印條碼；

G - 線條

語法：**G [:name;]x,y,r;ge:settings[,options]**

G	=	線條指令
[:name;]	=	定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；
x	=	x 軸座標
y	=	y 軸座標
r	=	旋轉角度，0~359 度間隔 1°
ge	=	線條類型：
		L：直線
		R：矩形
		C：圓形
		橢圓形可用圓形指令定義
settings	=	特殊圖形設定
[+options]	=	附加參數
	,fill	=
	,shade	=
	,outline	=

G - 線條 C - 圓形

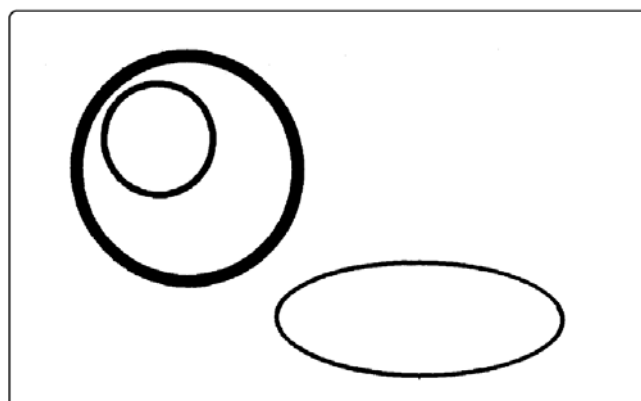
語法： **G** [:name;]x,y,r;**C**:radius1[,radius2[,width]][,options]

G = 線條指令
[:name;] = 定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；
x = （水平）x 軸座標
y = （垂直）y 軸座標
 圓形與橢圓形起始點在中心點
r = 旋轉角度
 在圓形圖案下無影響，在橢圓形圖案下則會有影響
C = 圓形
radius1 = 水平半徑
radius2 = 垂直半徑
width = 圓形線條的寬度
 如未設定寬度，則會印出實心圓或實心橢圓

[+options] = ,**fill** = 線條為實心模式（參見後面解說）
 ,**shade** = 線條為陰影模式（參見後面解說）
 ,**outline** = 線條有框線模式（參見後面解說）

範例：

```
J
S 11;0,0,68,71,100
G 65,50,0;C:25,10,.7
G 25,25,0;C:20,20,2
G 20,20,35;C:10,10,1
A 1
```



G - 線條 L - 直線

語法：**G [:name;]x,y,r;L:length,width[,start[,end]][,options]**

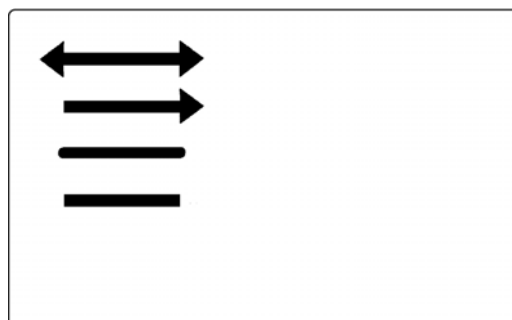
G = 線條指令
[:name;] = 定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；
x = （水平）x 軸座標
y = （垂直）y 軸座標
 直線起始點在該直線起始位置的中心
r = 旋轉角度，0~359 度間隔 1°
L = 直線
length = 長度
width = 寬度
start / end = 線條起點型態 / 線條終點型態
 s = 直角
 r = 圓弧
 a = 箭頭

如未設定線條起點/終點型態，則會以直角型態列印

[+options] = ,fill = 線條為實心模式（參見後面解說）
 ,shade = 線條為陰影模式（參見後面解說）
 ,outline = 線條有框線模式（參見後面解說）

範例：

```
J
S 11;0,0,68,71,100
G 5,5,0;L:24.5,2.5,a,a
G 5,15,0;L:24.5,2.5,s,a
G 5,25,0;L:24.5,2.5,r,r
G 5,35,0;L:24.5,2.5
A 1
```



G - 線條 R - 矩形

語法：**G [:name;]x,y,r;R:width,height[,hlt[,vlt]][,options]**

G = 線條指令

[:name;] = 定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；

x = （水平）x 軸座標

y = （垂直）y 軸座標

矩形起始點在左上角、矩形的外邊

r = 旋轉角度

R = 矩形

width = 矩形（水平）寬度

height = 矩形（垂直）長度

hlt = 水平線條厚度

vertw = 垂直線條厚度

如未設定寬度，則會印出實心矩形

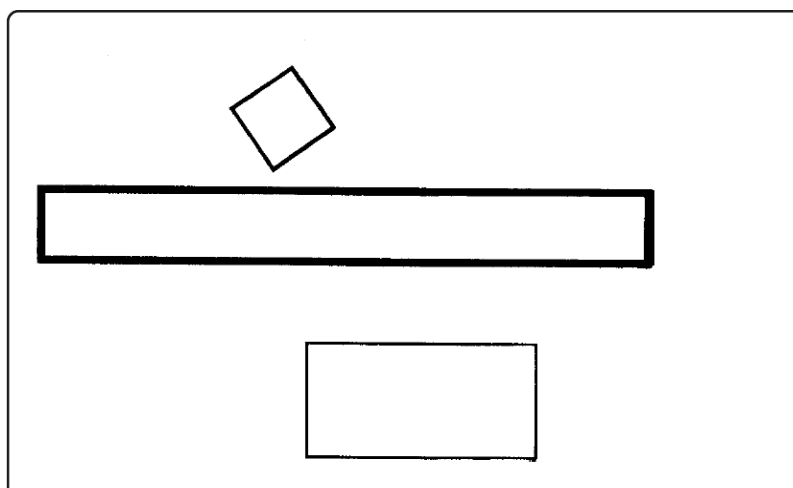
[+options] = **,fill** = 線條為實心模式（參見後面解說）

,shade = 線條為陰影模式（參見後面解說）

,outline = 線條有框線模式（參見後面解說）

範例：

```
J
S 11;0,0,68,71,100
G 35,45,0;R:30,15,.3,.3
G 0,25,0;R:80,10,1,1
G 25,15,35;R:10,10,.5,.5
A 1
```



G -線條 附加參數：Fill – 填滿

語法：**G [:name;]x,y,r;ge:settings[F:options]**

F = 填滿

options = 填滿模式參數，有效參數如下：

0%、6%、12%、25%、38%、50%、100%（網點密度）

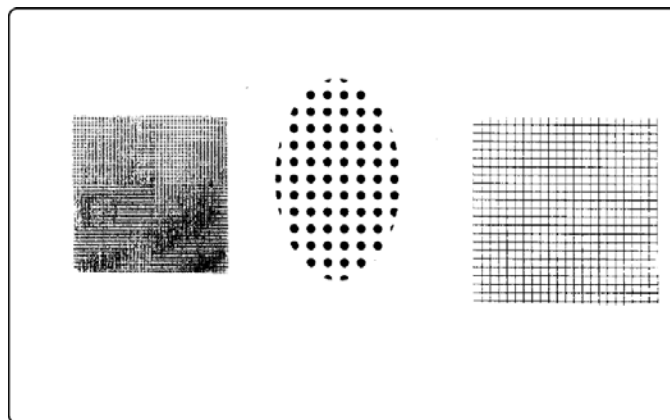
已定義模式有：

左(left)、右(right)、網點(dots)、網格(grid)、菱形(diamond)

圖形 1~4(user1~4，下載圖檔為 32x32 dots)

範例：

```
J
S 11;0,0,68,71,100
G 70,20,0;R:30,30, 1,20[F:grid]
G 48,30,0;C:10,16,10,10[F:dots]
G 5,20,0;R:25,25, 1,20[F:25%]
A 1
```



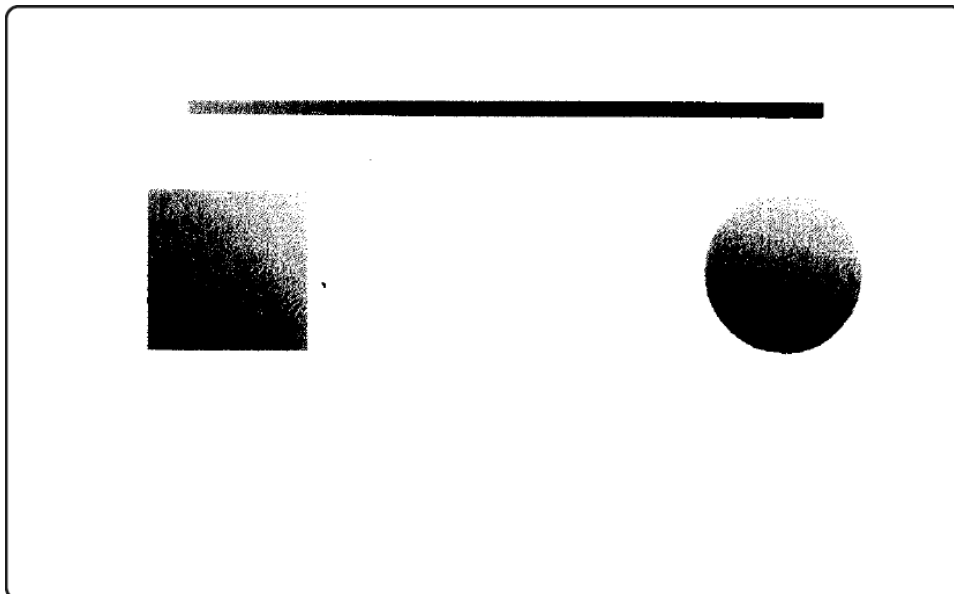
G -線條 附加參數：Shade – 陰影

語法：**G [:name;]x,y,r;ge:settings[S:%1[,%2[,direction]]]**

S = 陰影
 %1=黑色初始濃度值
 %2=黑色結束濃度值
 方向 direction=陰影角度

範例：

```
J
S 11;0,0,68,71,100
G 5,20,0;R:20,20, 1,20[S:60,10,45]
G 85,30,0;C:10,10,10,10[S:60,10,75]
G 10,10,0;L:80,2[S:30,90,0]
A 1
```



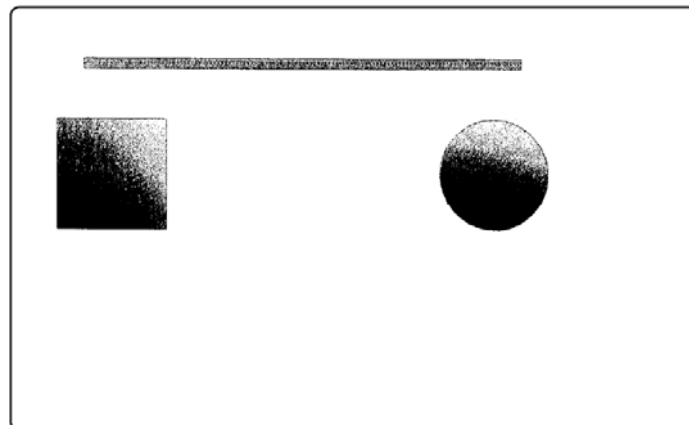
G - 線條 附加參數：Outline – 外框

語法： **G [:name;]x,y,r;ge:settings[shade options][O]**

O = 外框，用來呈現未填滿物體的外圍框線

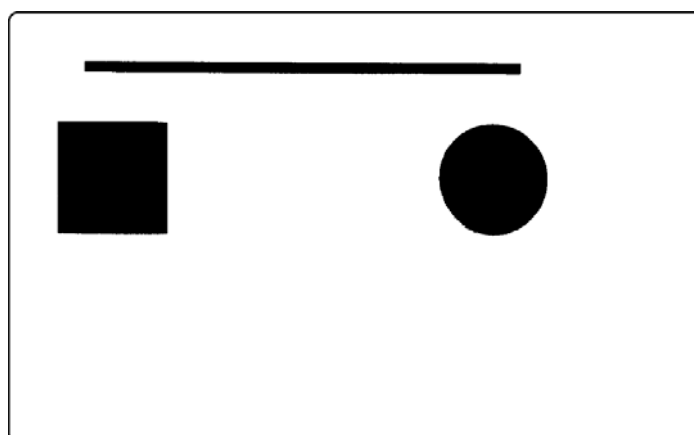
範例：

```
J
S 11;0,0,68,71,100
G 5,20,0;R:20,20,1,20[S:60,10,45] [O]
G 85,30,0;C:10,10,10,10[S:60,10,75] [O]
G 10,10,0;L:80,2[S:30] [O]
A 1
```



範例：

```
J
S 11;0,0,68,71,100
G 5,20,0;R:20,20,1,20 [O]
G 85,30,0;C:10,10,10,10 [O]
G 10,10,0;L:80,2 [O]
A 1
```



H - 列印溫度、速度、與列印方式

語法：**H speed[,h][,t][,r][,Bb]**

H = 打印頭工作溫度等級與列印速度指令

speed = 列印速度，各機型的最快列印速度不同，請先參照操作手冊，以便對列印速度能有正確設定，如設定錯誤，則會在條碼機的 LCD 控制面版上顯示錯誤設定值

h = 打印頭工作溫度等級（-20 到+10）

t = 打印型式，**T** 為熱轉列印，**D** 為熱敏列印，預設值為熱轉列印（**T**）

b = 回紙速度

範例：

H 150,0,D

說明：此設定為列印速度是 150mm/s，打印頭工作溫度等級為 0，以熱敏方式列印。

I - 圖片

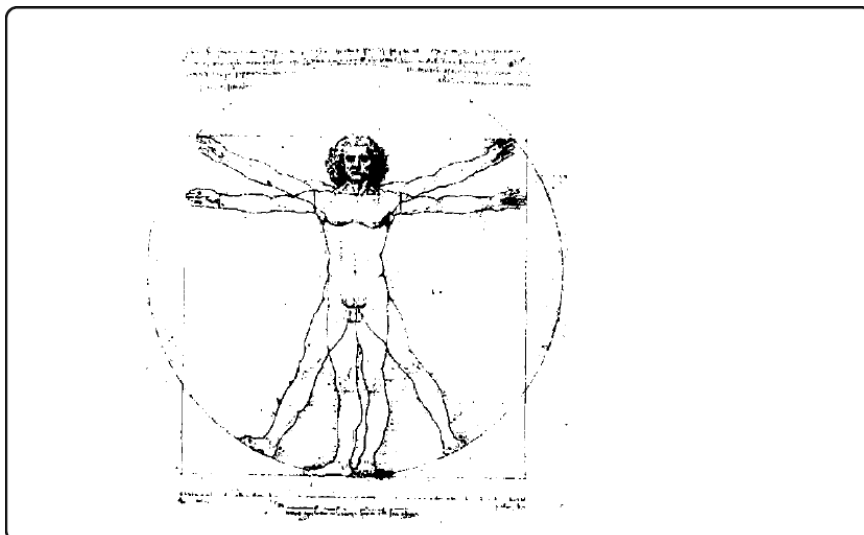
語法：I [:name;] x,y,r[mx,my];name

- I** = 圖片指令
- [:name;]** = 定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；
- x** = 圖片（水平）x 軸座標
- y** = 圖片（垂直）y 軸座標
- 最大圖片座標依條碼機機型不同而有所不同，請參照操作手冊
- r** = 旋轉角度，為 0°、90°、180°、與 270°
- mx** = 水平放大參數，數值為 1-10
- my** = 垂直放大參數，數值為 1-10

範例：

```
J
S 11;0,0,68,71,100
I:IMAGE1;20,5,0,0,0;HUMAN
T 12,25,0,3,6;Today's date is: [DATE:+03,+02,+10]
A1
```

說明：列印已下載到條碼機內部記憶體的 "HUMAN" 圖片，亦可將圖片儲存在記憶卡，再從程式中呼叫出來列印



J - 列印溫度、速度、與列印方式

語法：J [comment]

J = 列印開始指令

comment = 會顯示在條碼機 LCD 控制面版上的英文文字，
可當作標籤名使用，最大顯示長度為 15 個字元

範例：

J Adress label

說明：條碼機的 LCD 上會顯示“Adress label”，
並開始執行 J 後面的列印指令

M - 記憶卡存取

語法：**M variation...**

M 指令為條碼機在使用 CF 記憶卡時，對記憶卡做存取的動作

語法：**Mc [pathname]**

Mc = 記憶卡內容查詢

pathname = /card/ 呼叫記憶卡儲存的内容，此為預設值

= /iffs/ 呼叫條碼機內部快閃記憶體檔案系統内容

語法：**Md type;name**

Md = 刪除記憶卡資料

type = LBL (label, 標籤型式)

FNT (font, 字型型式)

IMG (image, 圖片型式)

FMT (label format, 標籤格式)

name = 記憶卡內的檔名

“type”:FNT 刪除所有 TTF 與 Speedo 字型

“type”:IMG 刪除所有指定檔名的圖片

範例：

```
M d IMG;logo
```

說明：刪除所有記憶卡裡的 “logo” 圖形檔，例如 logo.bmp 或 logo.pcx

語法：**Mf;name**

Mf = 格式化記憶卡

範例：

```
M f;MYDATA
```

說明：格式化記憶卡並將記憶卡命名為 “MYDATA”

語法：MI type;[pathname]name

MI = 讀取記憶卡資料

type = LBL (label, 標籤型式)

FNT (font, 字型型式)

IMG (image, 圖片型式)

FMT (label format, 標籤格式)

name = 記憶卡內的檔名

pathname = /card/ 呼叫記憶卡儲存的内容，此為預設值

= /iffs/ 呼叫條碼機內部快閃記憶體檔案系統内容

範例：

```
M1 LBL;TESTLBL
```

```
A2
```

說明：讀取記憶卡的 TESTLBL 標籤檔，並列印 2 張標籤

語法：Mr

Mr = 重複上一個檔案内容

範例：

```
Ms LBL;LOOP
```

```
J
```

```
S 11;0,0,68,70,100
```

```
T:Text1;20,10,0,3,7;[?:SerialNo:]
```

```
A3
```

```
Mr
```

```
Ms LBL
```

說明：將“LOOP”標籤檔儲存於條碼機的記憶卡，然後執行列印動作，條碼機會先在 LCD 上顯示“SerialNo”，並等待操作者輸入序號後，會列印 3 張標籤，然後再回到等待操作者輸入序號的狀態，直到操作者按下條碼機操控面版的取消鍵（CANCEL）才會取消整個列印動作

語法：Ms type;name

Ms = 儲存資料於記憶卡

type = LBL (label, 標籤型式)

FNT (font, 字型型式)

IMG (image, 圖片型式)

FMT (label format, 標籤格式)

name = 記憶卡內的檔名

pathname = /card/ 呼叫記憶卡儲存的內容，此為預設值

= /iffs/ 呼叫條碼機內部快閃記憶體檔案系統內容

範例：

```
Ms LBL;ASERIES
J
S 11;0,0,36,38,89
T:Text1;20,10,0,3,pt25;cab printers
A5
Ms LBL
```

說明：將“ASERIES”標籤檔儲存於條碼機的記憶卡，然後列印 5 張標籤，

如將標籤檔名設定為“DEFAULT.LBL”，則會在條碼機開機後便會立刻列印該標籤檔！

語法：Mu IMG;logo

Mu = 從記憶卡上傳二進制資料檔案內容

○ - 設定列印參數

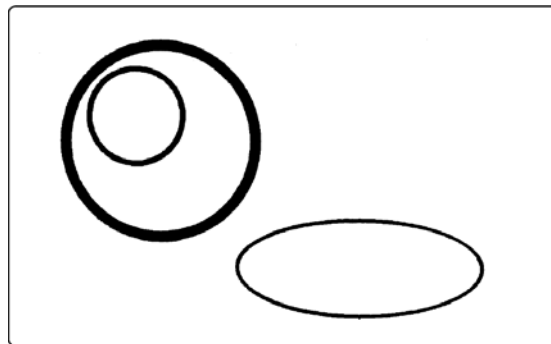
語法：○ [M,][R,][N,][T,][S,][U,][p,]

- = 列印參數指令
 - M = 標籤映象列印
 - R = 180°旋轉標籤方向
 - N = 整個標籤列印反白
 - S = 單一標籤暫存，當上一標籤完成列印時會處理下個標籤
 - T = 啟用撕紙模式，會在列印標籤後，
 - U = 封鎖暫停後重列印的功能，以避免同一標籤重複印兩次
 - p = 列印模式的回紙（backfeed）功能設定
- “P”指令會設定回紙功能為智慧型（smart），“D”指令則是啟動回紙（always），此參數設定會先暫時覆蓋條碼機硬體的設定，等該標籤列印工作完成後，條碼機才會回覆到原先回紙功能的設定模式。

O - 設定列印參數

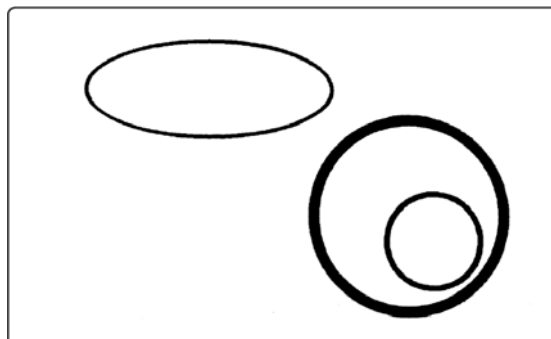
範例：

```
J
S 11;0,0,68,71,100
G 65,50,0;C:25,10,.7
G 25,25,0;C:20,20,2
G 20,20,35;C:10,10,1
A 1
```



範例：

```
J
O R
S 11;0,0,68,71,100
G 65,50,0;C:25,10,.7
G 25,25,0;C:20,20,2
G 20,20,35;C:10,10,1
A 1
```



說明：有使用跟無使用 **O R**（旋轉標籤）指令的列印情形

P - 設定標籤剝離模式或貼標模式

語法：**P [disp]**

P = 標籤剝離模式或貼標模式指令

disp = 標籤紙偏移位置，可用正數值與負數值

P 指令需放在 S 指令（定義標籤尺寸）後面！

S - 設定標籤尺寸

語法：**S [ptype;]xo,yo,ho,dy,wd[,dx,col][;name]**

- S** = 設定標籤尺寸指令
- ptype** = 設定標籤紙型式
 - e** = 標籤紙為連續紙，此參數會關閉條碼機的標籤紙感測器
 - 以下 **l** 為 **L** 的小寫
 - l0** = 標籤紙上方有反射標誌供條碼機的標籤紙感測器偵測
 - l1** = 標籤紙上有間距供條碼機的標籤紙感測器偵測
 - l2** = 標籤紙下方有反射標誌供條碼機的標籤紙感測器偵測
- xo** = 標籤起點水平 x 軸位移
- yo** = 標籤起點垂直 y 軸位移
- ho** = 列印方向的標籤高度
- dy** = 包含間距的標籤高度
- wd** = 標籤寬度
- 附加參數：當水平方向有多個標籤紙時
 - dx** = 水平方向的第一個標籤與第二個標籤之間的距離
 - col** = 水平方向的標籤數目，預設值為 1
 - name** = 顯示在條碼機 LCD 上的文字，
例如可用來顯示所要求放入的標籤紙規格

範例：

S l1;0,0,50,52,100

說明：定義標籤為有間距的標籤紙，高為 50mm、含間距的高為 52mm，寬為 100mm，水平與垂直位移為 0

標籤紙的最大寬度與最長高度設定會依不同條碼機而有所不同，此部分設定請參照條碼機使用手冊！

T - 文字

語法：**T [:name;]x,y,r,font,size[,effects];text**

- T** = 文字指令
- [:name;]** = 定義資料內容於唯一的資料名，供後續程式呼叫或其他用途（如數學運算）使用，最長 10 個字元，且不能使用特殊字元，其中資料名有字母大小之區別；
- x** = 水平起點 x 軸座標
- y** = 垂直起點 y 軸座標
- 直線起始點在該直線起始位置的中心
- r** = 文字旋轉角度，向量字型與 True type 字型皆可以間隔 1° 旋轉 360 度，Bitmap 自行則是可旋轉四個方向（0°、90°、180°、270°）
- font** = 以自行編號定義字型，可用條碼機內建字型或另外下載到條碼機的字型，下列字型為條碼機內建字型：

font nr.	Name	Type	Description
-1	_DEF1	Bitmap	Default-size 12x12 dots
-2	_DEF2	Bitmap	Default-size 16x16 dots
-3	_DEF3	Bitmap	Default-size 16x32 dots
-4	OCR_A_I	Bitmap	OCR-A Size I
-5	OCR_B	Bitmap	OCR-B
3	BX000003	Vector	Swiss 721™
5	BX000005	Vector	Swiss 721 Bold™
596	BX000596	Vector	Monospace 821™

- size** = 設定字型大小，向量字型可設定為毫米（mm）或英吋（inches），亦可設為 “pt x”；
bitmap 字型則是以 mx（水平，1-10 倍）與 my（垂直，1-10 倍）放大或縮小字型
- effects** = 字型特效，不同字型的特效亦不同，有效的字型特效如下：
- | | | |
|--------------------|-------------------|-----------------|
| b = 粗體 | s = 斜體 | z = 往左斜體 |
| n = 反白 | u = 下標 | i = 斜體 |
| k = 字體空隙最佳化 | v = 垂直列印文字 | |
- qn** = 壓縮字體，預設值為 100（不壓縮），可設定值 10-10000
- hn** = 字體寬度，h 亦可用 H
- mn** = 水平文字間距，n 為毫米（mm）或英吋（inches）
- 以下特效僅能用在條碼機內建 bitmap 字型
- o** = 外框（OCR 字型無效） **g** = 灰階（OCR 字型無效）

xn / yn = 水平 x 軸 / 垂直 y 軸放大倍數，n=1-10

text = 文字內容

T - 文字

以下為條碼機內建 Bitmap 字型實際列印情形，所有列印皆放大以便觀看：

FONT -1 (2x 2y)

Default Font 12x12 Dots

```
!@#$%^&*()_+|=\\<>?/[ ]';":{}
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
CüéÀâãäåæçèéëìíîïðñóôõöùü
ÛüéÿPfaíóúññ ò-¼½i«»
ÀÀÀ@ç¥ð
ædðéëëíííîíóððððù
úúúýý'-±q$÷,°".132
```

FONT -2 (2x 2y)

Default Font 16x16 Dots

```
!@#$%^&*()_+|=\\<>?/[ ]';":{}
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
CüéÀâãäåæçèéëìíîïðñóôõöùü
ÛüéÿPfaíóúññ ò-¼½i«»
ÀÀÀ@ç¥ð
ædðéëëíííîíóððððù
úúúýý'-±q$÷,°".132
```

FONT -3 (1x 1y)

Default Font 16x32 Dots

```
!@#$%^&*()_+|=\\<>?/[ ]';":{}
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
CüéÀâãäåæçèéëìíîïðñóôõöùü
ÛüéÿPfaíóúññ ò-¼½i«»
ÀÀÀ@ç¥ð
ædðéëëíííîíóððððù
úúúýý'-±q$÷,°".132
```

FONT -4

OCR A SIZE 1

```
!@#$%^&*()_+|=\\<>?/[ ]';":{}
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789
SSTZZ
P LA| "S<--Z
'*,>LRAAAALCCCEEEI
IDDNN0000RUUUUYT
/
```

FONT -5

OCR B

```
!@#$%^&*()_+|=\\<>?/[ ]';":{}
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789
SSTZZ
P LA| "S<--Z
'*,>LRAAAALCCCEEEI
IDDNN0000RUUUUYT
/
```


T - 文字

範例：

```
J
S 0,0,68,71,100
T 10,10,0,3,5;Font 3: Swiss
T 10,20,0,3,5;Font 3: S Bold
T 10,30,0,3,5,u;Font 3: Swiss Underline
T 10,40,0,3,5,s;Font 3: Swiss Slanted
T 10,50,0,3,5,n;Font 3: Swiss Reverse
T 10,60,0,5,5,s,u,n;Font 3: Swiss combined effects
A 1
```

Font 3: Swiss

Font 3: SBold

Font 3: Swiss Underline

Font 3: Swiss Slanted

Font 3: Swiss Reverse

Font 3: Swiss combined effects

X- 周邊訊號設定同步化

語法：**X y[;ao]**

- X** = 同步化外接裝置指令
- y** = 如應設定訊號時，列印座標
- ao** = 16 進制半字節設定或重置訊號

X 指令需放在 **S**（標籤尺寸設定）指令之後

此指令之功能與設定會取決於條碼機機型與周邊裝置，細節請參照操作手冊

時間：小時：分：秒

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;It is [H12] [MIN] [SEC]
A1
```

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,6;Actual time is [H024]:[MIN]:[SEC]
A1
```

說明：列印當時的時間，[H12] 為 12 小時制的小時，[H024] 為 24 小時制的小時，[MIN] 為分，[SEC] 為秒，該時間資料是從條碼機取得

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;It is [H012] o'clock
A1
```

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,9;The actual hour is [H024]
A1
```

說明：[H012] 為 12 小時制的小時，[H024] 為 24 小時制的小時，兩個指令主要會在 1~9 小時補上 0，即列印出 01~09；

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,8;The time is [TIME]
A1
```

The time is 23:08:57

時間：am/pm

語法：[XM]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,8;The time is [H12]:[MIN] [XM]
A1
```

The time is 7:16 am

時間：日期

語法：[DATE]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,5;Todays date is: [DATE]
A1
```

Todays date is: 10/11/2003

範例：

```
J
S 11;0,0,68,71,100
T 12,30,0,3,7;Date: [DAY02] - [MONTH02] - [YYYY]
A1
```

Date: 05-11-2003

說明：列印當時日期：日/月/年，其中 [DAY02] 與 [MONTH02] 設定為顯示 2 位數，不足 2 位數則補 0

時間：日期 - 加入一段時間

語法：[DATE{:+DD{,+MM{+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,6; Best before: [DATE:+03,+02,+10]
A1
```

Best before: 13/01/2014

說明：當日日期原為 10/11/2003，加上 3(日)/2(月)/10(年)後，日期變為 13/01/2014，適合用在保存期限的日期使用

時間：列印一年的第幾天

語法：[DOFY{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 12,20,0,3,7;February 5 is the
T 12,30,0,3,7;[DOFY] th day of the year
A1
```

February 5 is the
036 th day of the year

說明：當時日期為 2004 年 2 月 5 日，那天為一年的第 36 天，以三位數顯示為 036；

時間：列印週數(1-53)

語法：[WEEK{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,5;Date: [DATE]
T 12,35,0,3,5;This week is week number: [WEEK02]
A1
```

5/02/2004

This week is week number:06

說明：列印當時日期，並列印當時日期為該年的第 6 週，[WEEK02] 會以 2 位數顯示週數；

時間：列印完整或部分星期名

語法：[wday{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,5;The name of today is [wday]
T 12,35,0,3,5;In 2 days we have [wday:+02,00,00]
A1
```

The name of today is Thursday
In 2 days we have Saturday

說明：顯示當天為星期四（Thursday）及兩天後為星期六（Saturday）

語法：[wday2{:+DD{,+MM{,+YY}}}]

語法：[wday3{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 12,25,0,3,5;The name of today is [wday2]
T 12,35,0,3,5;In 2 days we have [wday2:+02,00,00]
A1
```

The name of today is Th
In 2 days we have Sa

說明：[wday2] 以 2 個字元顯示星期名，範例中日期為星期四（Thursday）則顯示為 Th，星期六顯示為 Sa，亦可以 [wday3] 顯示 3 個字元的星期名，但無顯示 4 個字元或更多字元的星期名；

時間：列印以數字表示星期名

範例：

```

J
S 11;0,0,68,71,100
T 12,25,0,3,5;The name of today is [WDAY]
T 12,35,0,3,5;In 2 days we have [WDAY:+02,00,00]
A1

```

The name of today is 4
In 2 days we have 6

說明：顯示以 4 代表當天為星期四，及兩天後為星期六，以 6 代表
0 = Sunday 星期日, 1 = Monday 星期一, 2 = Tuesday 星期二,
3 = Wednesday 星期三, 4 = Thursday 星期四, 5 = Friday 星期五,
6 = Saturday 星期六

時間：列印完整或部分月份名

語法：[mon{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,10;[mon]
A1
```



Feb

說明：列印以三個字元顯示當時月份為 **Feb**（二月）

語法：[month{:+DD{,+MM{,+YY}}}]

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,10;[month]
A1
```



February

說明：列印完整顯示當時月份為 **February**（二月）

時間：列印以數字表示月份名

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,8;[month] is Month [MONTH]
A1
```

February is Month 2

說明：列印當時月份及以 2 位數顯示當時月份

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,8;[month] is Month [MONTH02]
A1
```

February is Month 02

說明：列印當時月份及以 2 位數顯示當時月份，月份數不足 2 位數補 0

時間：列印年份

語法：[YY{:+DD{,+MM{,+YY}}}]

YY = 2 位數顯示年份（0-99）

語法：[YYYY{:+DD{,+MM{,+YY}}}]

YYYY = 4 位數顯示年份（1970-2069）

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,8;[month] - [YY]
A1
```

February-04

範例：

```
J
S 11;0,0,68,71,100
T 10,30,0,3,8;[month] - [YYYY]
A1
```

February-2004

加法運算：

語法：**[+:op1,op2, . . .]**

範例：

```
J
S 11;0,0,68,71,100
T:var1;25,10,0,3,5;44,80
T:var2;20,20,0,3,5;+
T:var2;25,20,0,3,5;26,70
G 20,25,0;L:20,0.3
T:res;25,35.0,.0,3,5.0;[+:var1,var2]
A1
```

44,80
+ 26,70
<hr/>
71.50

加法運算：

語法：`[-:op1,op2, . . .]`

範例：

```
J
S 11;0,0,68,71,100
T:var1;25,10,0,3,5;44,80
T:var2;20,20,0,3,5;-
T:var2;25,20,0,3,5;26,70
G 20,25,0;L:20,0.3
T:res;25,35.0,.0,3,5.0;[-:var1,var2]
A1
```

44,80
- 26,70
<hr/>
18.09

乘法運算：

語法：[*:op1,op2, . . .]

範例：

```
J
S 11;0,0,68,71,100
T:var1;25,10,0,3,5;44,80
T:var2;20,20,0,3,5;*
T:var2;25,20,0,3,5;26,70
G 20,25,0;L:20,0.3
T:res;25,35.0,.0,3,5.0;[*:var1,var2]
A1
```

44,80
* 26,70
<hr/>
1196.15

除法運算：

語法：`[/:op1,op2, . . .]`

範例：

```
J
S 11;0,0,68,71,100
T:var1;25,10,0,3,5;72
T:var2;20,20,0,3,5;/
T:var2;25,20,0,3,5;6
G 20,25,0;L:20,0.3
T:res;25,35.0,.0,3,5.0;[/:var1,var2]
A1
```

$$\begin{array}{r} 72 \\ / 6 \\ \hline 12.00 \end{array}$$

餘數運算：

語法：[%:op1,op2, . . .]

範例：

```
J
S 11;0,0,68,71,100
T:var1;25,10,0,3,5;84
T:var2;25,20,0,3,5;8
G 20,25,0;L:20,0.3
T:res;25,35.0,.0,3,5.0;[%:var1,var2]
A1
```

84
8
—
4.00

說明：列印 84 除以 8 的餘數為 4

範例：

```
J
S 11;0,0,68,71,100
T:COUNT;5,10,,3,4;[SER:000000][I]
T:MODCALC;5,10,,3,4;[%:COUNT,15][I]
T:SHIFT; 5,10,,3,4;[+:MODCALC,1][D:2,0]
A 20
```

說明：此程式產生計數器，從 1 累加到 15 然後跳回 1，列印 20 張

金額格式 – [P:…]

語法：[P:name,td{o}]

P = 金額格式指令
name = 金額格式指令
t = 千位數分隔號
d = 小數點與個位數分隔號
o = 附加補遺字元

範例：

```
J
S 11;0,0,68,71,100
T:Price1;10,20,0,3,8; [P:5432,.,-] [U:$20AC]
A 1
```

5.432,- €

說明：列印金額為 5432 歐元的內容，其中 [U:\$20AC] 為轉換成歐元符號 **€**，千位數分隔號為 “.”，小數點與個位數分隔號為 “，”，金額與歐元符號間的區隔符號為 “-”

小數點進位法 – [R:x]

語法：[R:x]

R = 小數點進位法指令
x = n：無進位法則（預設值）
x = u：無條件進入
x = d：無條件捨去
x = m：四捨五入

範例：

```
J
S 11;0,0,68,71,100
T 10,10,0,3,6;[*:5.191,5] [R:u]
T 10,20,0,3,6;[*:5.1898,5] [R:d]
T 10,30,0,3,6;[*:5.1898,5] [R:m]
A 1
```

25.96

25.94

25.95

LCD 提示 – [?:...]

語法：[?:x,y,z{,D}{,Lx}{,Mx}{,R}{,J}]

? = LCD 提示指令

x = 顯示在條碼機 LCD 上的文字

y = 預設等待輸入的資料值，如無設定則為空白，或以前次輸入的資料為主

z = 設定輸入的頻率

其他參數：

D = 刪除前次輸入的資料

Lx = 輸入資料長度，x=1-200（字元）

Mx = 設定輸入資料為下列定義：

- | | | | |
|----------|---|---|-------------|
| x | = | 0 | 具小數點分隔符號的數目 |
| | | 1 | 數值 0-9 |
| | | 2 | 小寫英文字母 |
| | | 3 | 小寫英文字母與數字 |
| | | 4 | 大寫英文字母 |
| | | 5 | 大寫英文字母與數字 |
| | | 6 | 大小寫英文字母 |
| | | 7 | 大小寫英文字母與數字 |
| | | 8 | 所有字元 |
| | | 9 | 符號與小數點分隔點 |

如在 **M** 指令後直接使用驚嘆號“！”，則不能使用空白字元！

R = 如無法從資料庫找到輸入的紀錄，則會重複提示輸入資料

J = 當條碼機要求輸入標籤數目時重複提示，如搭配 (A[?,R]) 指令可當簡單迴圈功能使用

範例：

[?:article number]

說明：在條碼機 LCD 上顯示要求輸入料號（article number）

範例：

[?:article number,7733214]

說明：在條碼機 LCD 上顯示要求輸入料號（article number），預設值為 7733214

範例：

[?:article,screw,3]

說明：在 LCD 上顯示要求輸入品名 (article)，預設值 screw，列印 3 張後跳回等待下一個輸入品名的狀態

範例：

[?:article no:,7733214,3,D]

說明：LCD 上顯示要求輸入料號，預設值為 7733214，每印三張標籤後清除前次輸入的資料 (即 7733214)，因此預設值只有在第一次有效

範例：

[?:article,screw,,L8]

說明：在 LCD 上顯示要求輸入品名 (article)，預設值 screw，最大輸入資料長度為 8 字元/數值

範例：

[?:number,7733214,,M1111111]

說明：LCD 上顯示要求輸入號碼，預設值為 7733214，並設定只能輸入 7 位數數值

範例：

[?:artno?,,1,M1114444]

說明：LCD 上顯示要求輸入料號 (artno?)，無預設值，且設定只能輸入 7 個字元，前三位為數值，後四位為大寫英文字母

範例：

[?:article?,,1,M1111111,R,D]

說明：LCD 上顯示要求輸入品名 (article?)，無預設值，且設定只能輸入 7 個數值，如無法從資料庫裡找到記錄則會重複該提示，並刪除前次輸入的資料

範例：

```
[?:article,2200333,,,L6,M!11111]
```

說明：LCD 上顯示要求輸入品名，預設值為 2200333，且設定只能輸入 5 個無空白的數值，即使預設值 2200333 為 7 位數，及長度限制為 6 位，實際操作時，也只會顯示 5 位數（M! 11111）

範例：

```
J simple loop
S 11;0,0,68,71,100
T 10,15,0,3,10;[SER:1]
T 10,30,0,3,10;[?:INPUT?]
T 10,45,0,3,10;[?:Second INPUT?,,,J]
A [?,R]
```

說明：此例為簡易迴圈的應用，[SER:1] 跳序號，從 1 起跳，[?:INPUT?] 為僅一次提示輸入，[?:Second INPUT?,,,J] 則會重複提示輸入，直到按取消鍵（CANCEL）為止

置換 0 – [C:....]

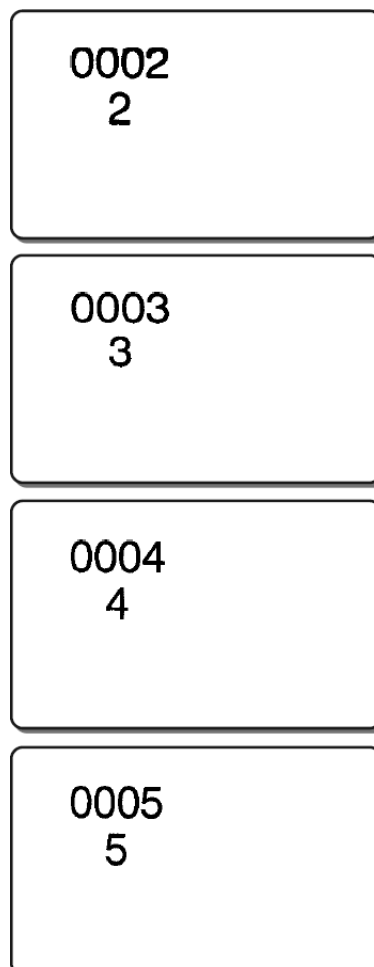
語法：[C:fill{,base}]

C = 置換左邊的 0
fill = 填滿字元
base = 設定計數系統的附加參數

範例：

```
J
S 11;0,0,68,71,100
T:CNT; 10,15,0,3,10;[SER:1][I]
T:FIELD1;10,10,0,3,10;[+:1,CNT][C:0][D:4,0]
T:FIELD2;10,20,0,3,10;[+:1,CNT][C: ][D:4,0]
A 5
```

說明：列印 5 張標籤，標籤內有 2 組計數器，從 2 起跳，[D:4,0] 設定 4 位數值，[C:0] 指令設定要補 0，故第一個計數器有補 3 個 0，第二個計數器是 [C:]，故不補 0



位數設定 – [D:....]

語法：[D:m,n]

- D** = 設定顯示位數
- m** = 總位數
- n** = 小數點後面的位數，預設值為 2 位

範例：

```
J
S 11;0,0,68,71,100
T:input;10,30,0,3,14;[*:10.79,4.16] [D:4,2]
A 1
```

44.88

資料庫檔案連結 – [DBF:....]

語法：[DBF:key,keyvalue,entry]

DBF	=	資料庫檔案連結指令
key	=	搜尋資料庫的資料
keyvalue	=	資料庫記錄的資料值定義為英文字母及數字
entryfield	=	傳送記錄的資料值

範例：

[DBF:NUMBER, NUMBERTA, ARTICLE]

說明：搜尋資料庫裡的 NUMBERTA 資料表裡的 NUMBER 資料值，
並傳送 ARTICLE 數值；

此指令需搭配 E 指令使用，一張標籤格式僅能使用一個資料庫，此指令功能僅適用於小型資料庫，大型資料庫則建議使用 cab Database Connector 軟體才有好的效能。

隱藏內容 – [I]

語法：[I]

範例：

```
J
S 11;0,0,68,71,100
T:WEIGHT;10,20,0,3,5;[?:Weight?][I]
T:PRICEUNIT;10,20,0,3,5;[I] 2.65
T:RESULT;10,40,0,3,4;The Fish price is: [*:WEIGHT,PRICEUNIT]
A 1
```

說明：此範例會在條碼機 LCD 上顯示要求分兩次輸入重量（Weight）與單價（PRICEUNIT），並將重量與單價相乘後列印出結果（RESULT），而重量與單價資料並不會列印在標籤上，下面列印結果是以魚（Fish）的重量為 12 公斤、單價為 2.65/公斤，結果為 31.79：

The Fish price is: 31.79

對齊 – [J:...]

語法：[J:m]

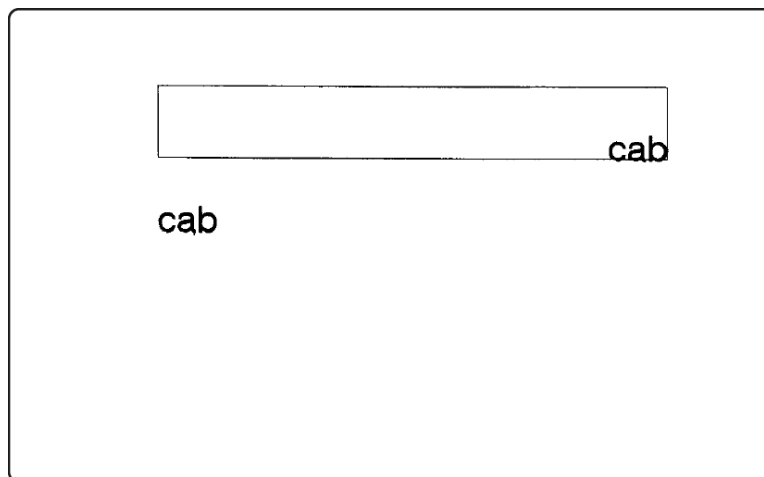
J = 對齊指令
m = **l**：靠左 (left)
 c：置中 (centered)
 r：靠右 (right)
l = 文字對齊位置的特別區域長度設定

範例：

```

J
S 11;0,0,68,71,100
G:AREA;10,10,0;R:70,10,.2,.2
T:NOADJUST;10,30,0,3,5;cab
T:ADJUST;10,20,0,3,5;cab[J:r70]
A 1
  
```

說明：在 NOADJUST 指令行裡的 cab 不會做對齊的動作，而 ADJUST 指令行的 cab 會做靠右（長度 70 位元）對齊



存取資料名 – [name]

語法：**[name]**

name = 定義資料內容於唯一的資料名，供後續程式呼叫使用，其中資料名有字母大小之區別；

範例：

```
J
S 11;0,0,68,71,100
T:FIELD1;10,20,0,3,5;cab
T:FIELD2;10,30,0,3,5;label printers
T:FIELD3;10,40,0,3,4;we like [FIELD1] [FIELD2]  !!
A 1
```

說明：將 cab 定義為 FIELD1，再將 label printers 定義為 FIELD2，最後將 FIELD1 及 FIELD2 與 we like!! 定義為 FIELD3 並印出 FIELD1、FIELD2 與 FIELD3：

```
cab
label printers
we like cab label printers  !!
```

複製字串 – [name,m{n}]

語法：[name,m{n}]

- name** = 定義資料內容於唯一的資料名，供後續程式呼叫使用，其中資料名有字母大小之區別；
- m** = 要複製字串的第一個開始複製字元位置
- n** = 要複製字串的字元總數

範例：

```
J
S 11;0,0,68,71,100
T:ORIGINAL;10,20,0,3,8;cab GERMANY
T:CUTOFF;10,40,0,3,8;[ORIGINAL,8,4]
A 1
```

說明：將 **cab GERMANY** 定義為 **ORIGINAL**，再將 **ORIGINAL** 的資料內容的第 8 個字元開始複製，共複製 4 個字元，即複製 **MANY**，然後將 **MANY** 定義於 **CUTOFF** 資料名，並列印出 **ORIGINAL** 與 **CUTOFF**：

cab GERMANY

MANY

讀取暫存檔 – [RTMP...]

語法：[RTMP]

語法：[RTMP:x]

RTMP	=	讀取 TMP 暫存檔
x	=	設定資料重複次數

序列號 – [SER : ...]

語法：[SER:start{,incr{,freq}}]

SER = 啟用序列號指令
start = 序列號初值；
incr = 序列號增加值
freq = 每隔幾張跳一次序列號

如未設定 **incr** 與 **freq**，則條碼機會自動以預設值 1 為設定；

範例：

```
J
S 11;0,0,68,71,100
T:CNT; 10,15,0,3,10;[SER:1][I]
T:FIELD1;10,10,0,3,10;[+:1,CNT][C:0][D:4,0]
T:FIELD2;10,20,0,3,10;[+:1,CNT][C: ][D:4,0]
A 4
```

說明：此例是序列號累加值為 1，但不列印序列號，而是將序列號定義為 CNT，並將 CNT 與 1 做加法運算，然後把運算值再定義為 FIELD1 與 FIELD2，再列印出 FIELD1 與 FIELD2；

0002
2

0003
3

0004
4

0005
5

Unicode 字元 – [U : x...]

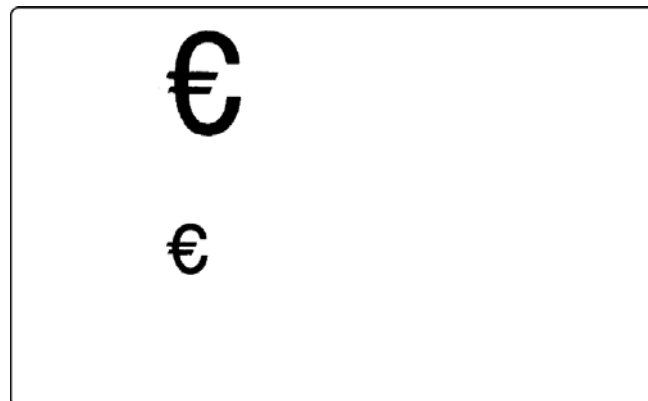
語法：[U:x]

- U** = 選用 Unicode 字元
- x** = 以 \$ 符號或 ASCII 控制碼名表示 16 進制值，如 NUL、SOH、STX、ETX、EOT、ENQ、ACK、BEL、BS、HT、LF、VT、FF、CR、SO、SI、DLE、DC1、DC2、DC3、DC4、NAK、SYN、ETB、CAN、EM、SU、ESC、FS、GS、RS 與 US，或 128 碼的控制碼：FNC1、CODEA、CODEB、CODEC；

範例：

```
J
S 11;0,0,68,71,100
T 20,15,0,3,20;[U:$20AC]
T 20,40,0,596,10;[U:$20AC]
A1
```

說明：[U : \$20AC]為歐元符號，另外，[U : FNC1]為設定 128 碼的碼 1 字元；



英文大小寫轉換 – [UPPER:...]、[LOWER:...]

語法：[UPPER:Name]

語法：[LOWER:Name]

UPPER = 將英文字母做大寫轉換；

LOWER = 將英文字母做小寫轉換；

範例：

```
J
S 11;0,0,68,71,100
T:Input;10,20,0,3,8;cab Germany
T:UPPERCASE;10,40,0,3,8;[UPPER:Input]
A 1
```

cab Germany

CAB GERMANY

範例：

```
J
S 11;0,0,68,71,100
T:Input;10,20,0,3,8;cab GERMANY
T:LOWERCASE;10,40,0,3,8;[LOWER:Input]
A 1
```

cab GERMANY

cab germany

寫入 LOG 檔 – [WLOG]

語法：**[WLOG]**

WLOG = 將資料寫入記憶卡的 LOG 檔，可讓列印標籤或記錄使用

範例：

```
E LOG;EXAMPLE  
T:VAL; 5,6,0,3,3;[SER:0001]  
T:PRINT;5,6,0,3,3;Label [VAL] printed at [DATE] um [TIME].[WLOG][I]
```

說明：此範例是將 VAL 計數值寫入 LOG 檔以便記錄與列印標籤數量

寫入暫存檔 – [WTMP]

語法：[WTMP]

WTMP = 將資料寫入條碼機記憶卡內之已定義的暫存檔；

範例：

```
E TMP;EXAMPLE  
T:XVAL;10,10,,0,3,3;[RTMP,3][I]  
T:SERNO;10,10,0,3,3;[+:XVAL,1][C:0][I][WTMP]  
T:TESTFELD;10,10,0,3,3;Serial number is:[SERNO]
```

說明：定義暫存檔名為 **EXAMPLE**，並將暫存檔的資料定義為 **XVAL**，及 **XVAL** 的值會以暫存檔 **EXAMPLE** 為依據；

cab 資料庫連結指令

注意：欲使用資料庫連結功能，必須安裝支援的硬體。

cab Database Connector（軟體）

該軟體可讓 cab A 系列條碼機及網路卡透過 TCP/IP 傳輸，列印包含 SQL 相容之資料庫的標籤，可經由條碼機的按鍵呼叫資料；

目前不使用 cab Database Connector 軟體的方式是，必須將儲存於記憶卡之固定格式的資料讀取到條碼機，此方式缺點是，該資料必須經過轉換，當資料越多，速度就越慢，因此並不實際，且改變核心資料庫時，則要更新條碼機上的記憶卡；

而使用 cab Database Connector 軟體的方式則不同，該軟體可從網路上的任何地方呼叫現存的資料庫，如列印新標籤時，在該資料庫異動的同時，該軟體也會立即更新資料；

使用資料庫連結時所需配備：

- A 系列條碼機
- 具備網路卡與支援 cab Database Connector 功能
- CF 記憶卡
- 輸入裝置（HL30 USB 掃描槍或 USB 鍵盤）
- cab Database Connector 軟體

cab A 系列條碼機都具 SQL 客戶端連結功能，可直接經由 cab Database Connector 軟體與 TCP/IP 網路與資料庫伺服器做連結，所有 ODBC 資料庫或微軟 OLEDB 介面都可支援；在 cab Database Connector 伺服器端上，都可同時取得數個 tables 與 fields，多個已定義的標籤可以由記憶卡上的內容表上選擇；

運作原理：

cab SQL 客戶端透過 TCP/IP 網路連接 cab Database Connector 並送出 SQL 查詢，cab Database Connector 接收到 SQL 查詢後會經由 ADO（ActiveX DATA Object）送出該查詢到資料庫伺服器端；

cab Database Connector 從資料庫伺服器端收到資料記錄後，會經由 TCP/IP 方式送出該資料記錄到 cab SQL 客戶端，條碼機的 SQL 客戶端就會收到原先要求的資料記錄；

支援的資料庫：

MS Access, Ms SQL Server, Oracle, Dbase 與 ODBC 連結

重要事項：必須安裝 Jet40Sp3_Comp.exe 與 mdac_typ.exe，通常安裝 Windows 2000 或 Office 2000 後，都會有這些檔案，如使用 Windows 98/95 時，則需要另外安裝這檔案，可在微軟網站（www.microsoft.com/data）下載。

cab Database Connector 與 A 系列條碼機之 SQL 客戶端功能

A 系列條碼機可由 cab Database Connector 軟體與內建的 SQL 客戶端功能透過 TCP/IP 網路直接從資料庫取得所需資料；當使用 A 系列條碼機單機操作功能（即不外接電腦）時，可不需再記憶卡上儲存與維護資料表；

使用者可連結所有具 ODBC 或微軟 ADO 介面的資料庫；

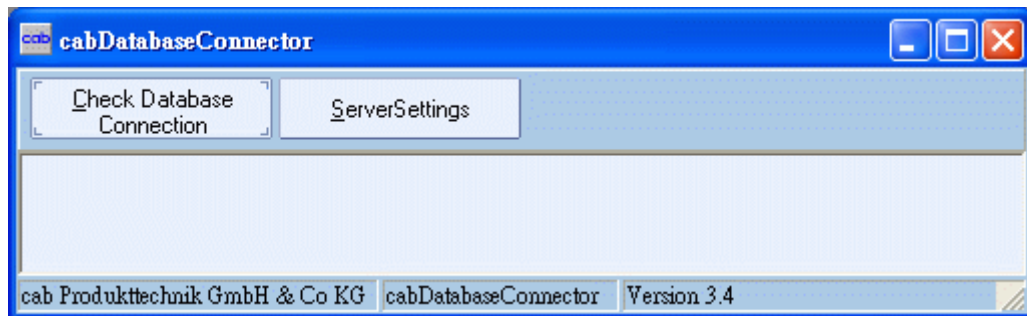
同時可連結數個資料表（table），且速度會比使用記憶卡連結資料還快上許多。

操作方式

步驟一

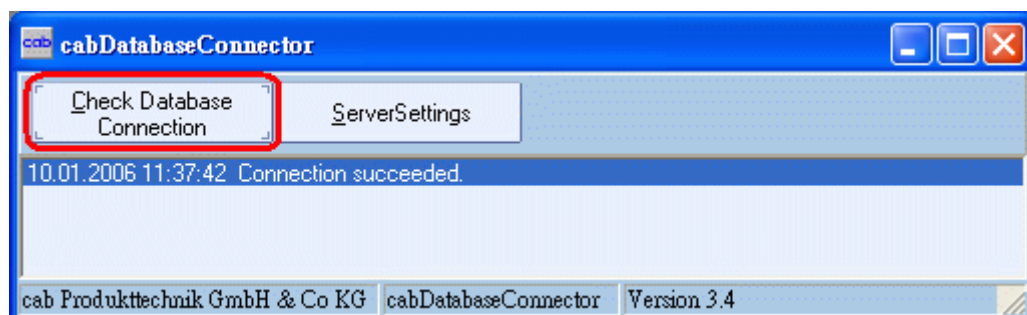
將 cabDatabaseConnector.exe 複製到電腦任意指定的目錄下，然後執行該軟體，cab Database Connector 並不需要任何 DLL 檔或其他程式，除非是系統本身所需的檔案（如重要事項裡所提到的檔案）；

執行軟體後會出現的視窗如下：



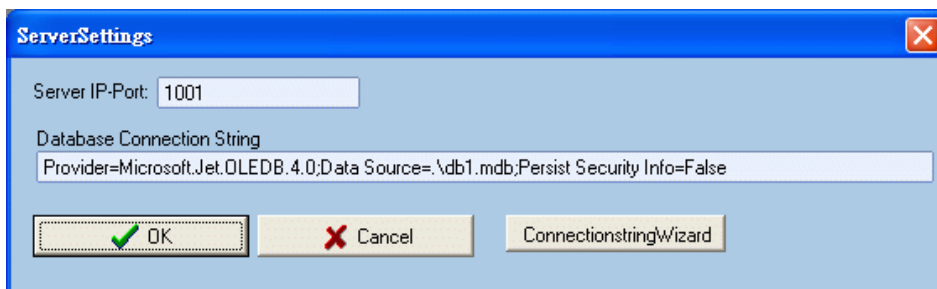
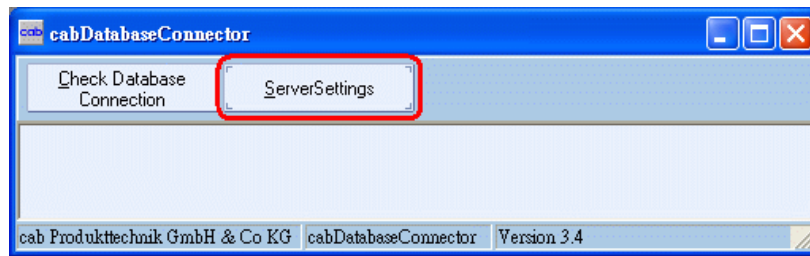
步驟二

先按下 Check Database Connection 檢測電腦與條碼機網路連結是否正常，正常則會出現連結成功的訊息：

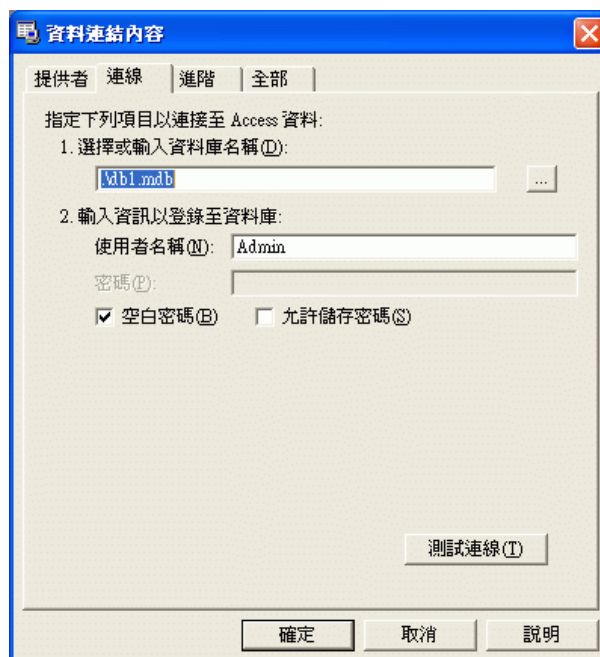
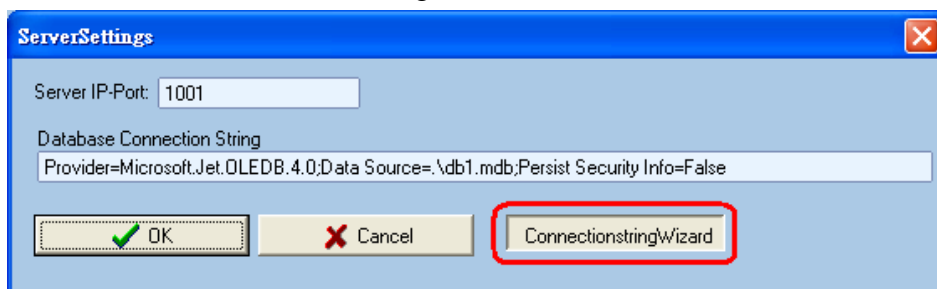


步驟三

按下 ServerSettings 鍵，可做資料庫連結的更改：



在 Database Connection String 下行裡，鍵入完整的資料庫連結路徑，或可使用連結路徑精靈（ConnectionstringWizard）做進階設定：



進階設定部分則需要對資料庫有基本的瞭解；

連結路徑範例：

MSAccess: Provider=Microsoft.Jet.OLEDB.4.0;Data-Source=<DatabasePath+MDB-Filename>

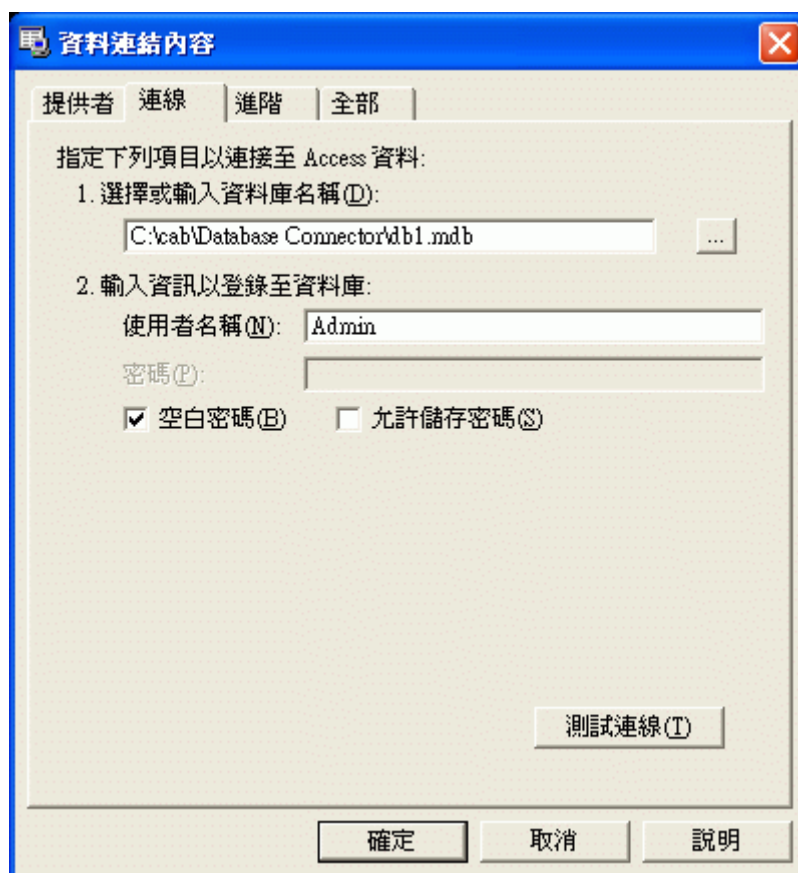
ODBC: in most cases simply type in the ODBC-Datasourcename

MSSQLServer: Provider=SQLOLEDB.1;Integrated Security=SSPI; Persist SecurityInfo=False;Initial Catalog=cab; Data Source=hostname

ORACLE: Provider=MSDAORA.1;User ID=User; Data Source=Prod;Persist Security Info=False

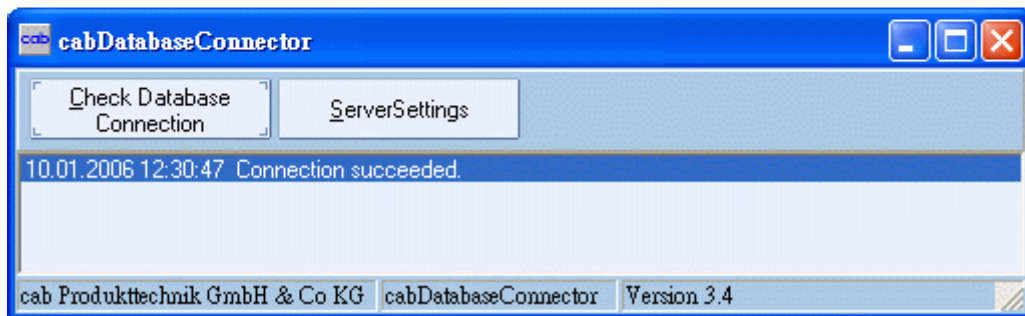
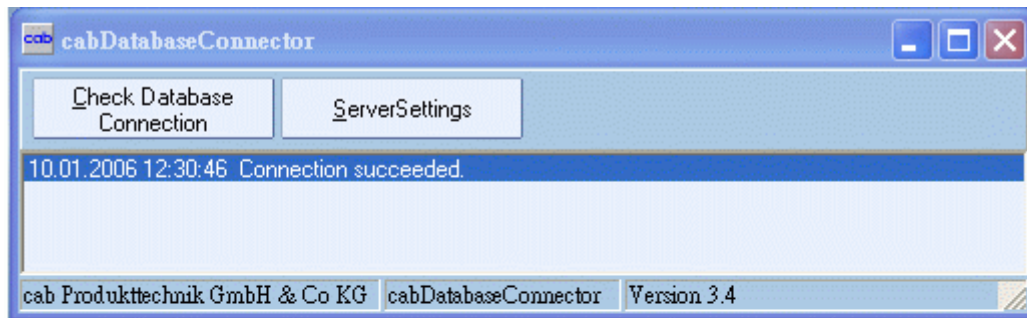
Dbase: DSN=ExampleDatasource;DBQ=<DatabasePath>; DefaultDir=<DatabasePath>;FIL=dBase IV

如以所附的範例檔案 (db1.mdb) 為例，則指向 cab Database Connector 目錄：

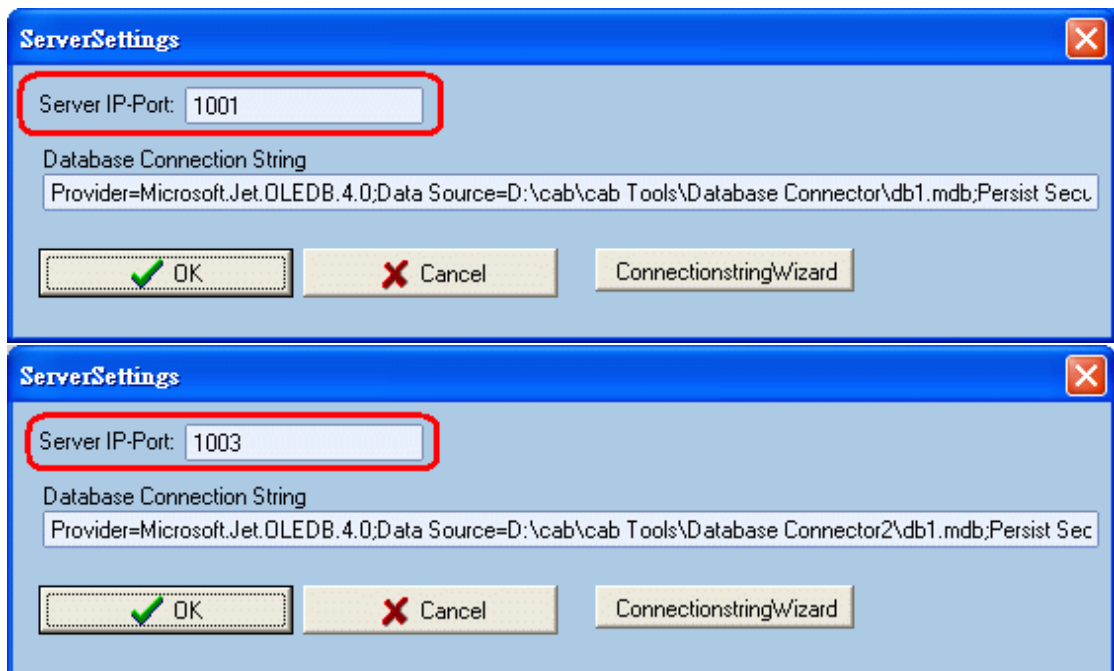


注意：

cab Database connector 可在一台電腦上開啓數個視窗做網路連結：



欲使用多連結時，僅需更改 IP-Port 即可：



更改方式是開啓 cab Database Connector 目錄裡的 cabDatabaseConnector.ini，

[IP]

Port=1001

1001

將 1001 改為其他可用的接口，如 1003 即可；

步驟四

將設計好的標籤檔儲存於 A 系列條碼機的記憶卡裡，在該標籤檔裡必須加入資料庫連結指令才能正確連結資料庫；

資料庫連結指令：**E**

語法：**E SQL；<IP of cabDatabase connector>：Portnumber**

說明：

IP 位址是設定 cab Database Connector 所在的電腦位址，portnumber 則是要與 cab Database Connector 設定的 Port 相同；

範例：**E SQL；192.168.0.80:1001**

說明：連接電腦的 IP 位址是 192.168.0.80，Port 為 1001

資料庫存取功能

語法：**[SQL:Select Field from Table where Searchvalue='{Fieldname}']**

說明：SQL 指令用於 SQL 資料庫存取資料

範例：

T 10,15,0,3,5;[SQL:SELECT PRODNAME FROM TA WHERE ARTICLE= '{ARTNR}']

SPLIT 指令

語法：**[SPLIT:Field,Index]**

範例：T 10,5,0,3,5;[SPLIT:RESULT,1]

下列為接下來的範例能成功連結資料庫的必要條件：

- A 系列條碼機有外接 USB 鍵盤
- 有安裝 CF 記憶卡
- A 系列條碼機需有安裝網路卡且支援資料庫連結功能
- cab Database Connector 有正確啟動與設置
- 使用的資料庫必須是有效的，範例使用的 table 名是 TA，field 名是 ARTICLE，其對應值是 “{ARTNR}”，PRODNAME 內容將會是由資料庫提供
- 下面標籤範例須儲存於記憶卡

範例：

行數	程式內容
1.	m m
2.	J
3.	S l1;0,0,68,70,100
4.	H 200
5.	E SQL;192.168.0.128:1001
6.	T:ARTNR;10,5,0,3,5;[?:Article Nummer,5560432,1,R,D]
7.	T 10,15,0,3,5;[SQL:SELECT PRODNAME FROM TA WHERE ARTICLE='{ARTNR}']
8.	A 3

說明：

行 1：設定長度單位（mm）

行 2：開始程式

行 3：設定標籤尺寸，此範例是標籤設定為寬度 100mm、高度 68mm

行 4：設定列印速度為 200 mm/s

行 5：設定連接電腦的 IP 為 192.168.0.128 及資料庫連結的 port 為 1001

行 6：定義標籤文字內容，並會顯示在條碼機 LCD 操控面板上（T:ARTNR...）

行 7：設定 SQL 請求與同時定義位置及資料區字型

行 8：設定列印張數為 3 張



希愛比科技股份有限公司
台北縣中和市中正路 700 號 9F-8

Tel : +886 (0) 2 82273966

Fax : +886 (0) 2 82273566

<http://www.cabd.de>

<http://www.cabasia.net>

email: cabasia@cab.de

copyright by cabAsia

All specifications about delivery, design, performance and weight are given to the best of our current knowledge and are subject to change without prior notice.