

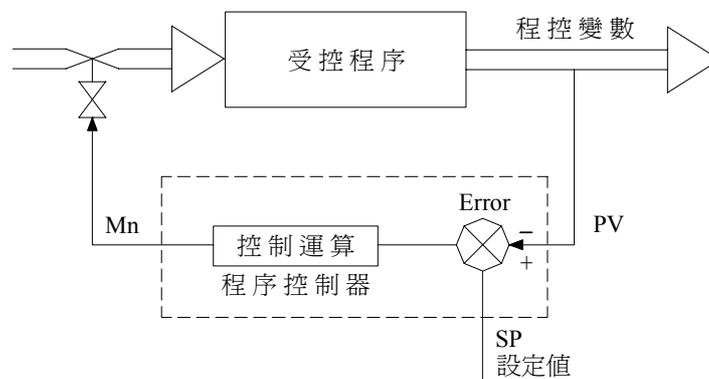
## 附錄：工程變更新增指令使用說明

### 第 21 章：FB-PLC 之 PID 功能與應用

#### 21.1 PID 控制簡介

一般常見之程序控制應用，只要控制元件加工精度夠或是控制反應之重現性夠好，開迴路控制已可滿足大部份之應用需求，而且簡單容易、成本低廉是最大優點；但隨著使用時間、元件特性變化或受控負載或外界工作環境之變異，開迴路控制因為沒有忠實將受控程序之實際量迴饋至控制器，因此控制結果可能與實際期望之結果會有些許落差，閉迴路 PID 程序控制是用來克服解決上述缺失的極佳選擇。

FB-PLC 提供軟體數位化之 PID 數學運算式，對於一般反應之閉迴路程序控制足可應付所需，但對於需快速反應之閉迴路控制欲使用本功能需事先評估是否適用。典型之閉迴路程控示意圖如下圖所示：



#### 21.2 控制器選擇

根據應用需求，使用者可將 PID 控器設定成比例式控制、比例+積分控制器或比例+積分+微分控制器；各類控制器之數學式與特性要說明如後。

##### 21.2.1 比例式控制器

數位化數學運算式如下：

$$Mn = (1000/Pb) \times (En) + Bias$$

Mn：「n」時之控制輸出量

Pb：比例帶（範圍：2~5000，單位為 0.1%；Kc(增益)=1000/Pb）

En：「n」時之誤差=設定值（SP）-「n」時之程控變數值（PVn）

Ts：比例運算之間隔時間（範圍：1~3000，單位：0.01S）

Bias：偏置輸出量（範圍：0~4095）

比例式控制器運算簡單省時，大部份之應用足可勝任；缺點是當設定值有變更時，必須調整偏置輸出量（Bias）以消除穩態誤差。

### 21.2.2 比例+積分控制器

數位化數學運算式如下：

$$Mn=(1000/Pb) \times (En) + \sum_0^n [(1000/Pb) \times Ti \times Ts \times En] + Bias$$

Mn : “n” 時之控制輸出量

Pb : 比例帶 (範圍：2~5000, 單位為 0.1% ; Kc(增益)=1000/Pb)

En : “n” 時之誤差=設定值 (SP) - “n” 時之程控變數值 (PVn)

Ti : 積分時間常數 (範圍：0~9999, 相當於 0.00~99.99 Repeats/Minute)

Ts : 比例+積分運算之間隔時間 (範圍：1~3000, 單位：0.01S)

Bias : 偏置輸出量 (範圍：0~4095)

加上積分項之控制器可以消除只有比例式控制器時所產生之穩態 Offset, 也就是說可以自動消除穩態誤差。偏置輸出量 (Bias) 可以為 0。

### 21.2.3 比例+積分+微分控制器

數位化數學運算式如下：

$$Mn=(1000/Pb) \times (En) + \sum_0^n [(1000/Pb) \times Ti \times Ts \times En] - [(1000/Pb) \times Td \times (PVn - PVn-1) / Ts] + Bias$$

Mn : “n” 時之控制輸出量

Pb : 比例帶 (範圍：2~5000, 單位為 0.1% ; Kc(增益)=1000/Pb)

En : “n” 時之誤差=設定值 (SP) - “n” 時之程控變數值 (PVn)

Ti : 積分時間常數 (範圍：0~9999, 相當於 0.00~99.99 Repeats/Minute)

Td : 微積分時間常數 (範圍：0~9999, 相當於 0.00~99.99 Minute)

PVn : “n” 時之程控變數值

PVn-1 : “n” 時之上一次程控變數值

Ts : PID 運算之間隔時間 (範圍：1~3000, 單位：0.01S)

Bias : 偏置輸出量 (範圍：0~4095)

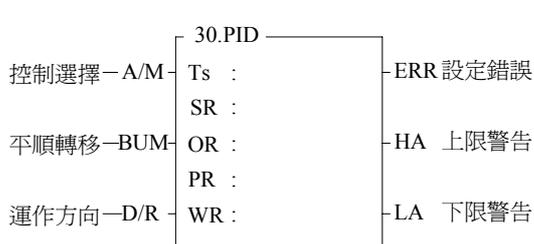
加上微分項之控制器, 目的在消除程控系統之過度反應, 進而使程控系統能夠平穩緩和達到穩定。

雖然微分項有上述優點, 但因其對輸出量之貢獻相當靈敏, 大部分之應用不必使用微分項而將 Td 設定為 0。

## 21.3 PID 指令說明與程式範例

本節將就 FUN30 (PID) 指令之使用作詳細說明與介紹並舉例說明。

FUN30 PID	PID 運算便利指令	FUN30 PID
--------------	------------	--------------



Ts : PID 運算間隔時間

SR : 程控設定值起始暫存器號碼，共佔用 8 個暫存器

OR : PID 輸出暫存器號碼

PR : 參數設定值起始暫存器號碼，共佔用 7 個暫存器

WR : 本指令所需使用之工作暫存器起始號碼，共佔用 5 個暫存器，其它地方不可重覆使用。

運算元	範圍	HR	ROR	DR	K
		R0   R3839	R5000   R8071	D0   D3071	
Ts	○	○	○	○	1~3000
SR	○	○	○*	○	
OR	○	○	○*	○	
PR	○	○	○*	○	
WR	○	○	○*	○	

● PID 指令(FUN30)係將目前所量測之外界類比輸入值當作程控變數 ( Process Variable, 簡稱 PV), 將使用者所設定之設定值 ( Setpoint, 簡稱 SP) 與程控變數經由軟體 PID 數學式運算後, 得到適宜之輸出控制值經由 D/A 類比輸出模組或再處理經由其它界面以控制受控程序在使用者所期望之設定範圍內。

● 數位化 PID 運算式如下：

$$M_n = (1000/P_b) \times (E_n) + \sum_0^n [ (1000/P_b) \times T_i \times T_s \times E_n ] - [ (1000/P_b) \times T_d \times (P V_n - P V_{n-1}) / T_s ] + Bias$$

- M<sub>n</sub> : "n" 時之控制輸出量
- P<sub>b</sub> : 比例帶 (範圍：2~5000, 單位為 0.1%; K<sub>c</sub> (增益) =1000/ P<sub>b</sub>)
- T<sub>i</sub> : 積分時間常數 (範圍：0~9999, 相當於 0.00~99.99 Repeats/Minute)
- T<sub>d</sub> : 微分時間常數 (範圍：0~9999, 相當於 0.00~99.99 Minutes)
- PV<sub>n</sub> : "n" 時之程控變數值
- PV<sub>n-1</sub> : "n" 之上一次之程控變數值
- E<sub>n</sub> : "n" 時之誤差=設定值 (SP) - "n" 時之程控變數值 (PV<sub>n</sub>)
- T<sub>s</sub> : PID 運算之間隔時間 (範圍：1~3000, 單位：0.01S)
- Bias : 偏置輸出量 (範圍：0~4095)

FUN30 PID	PID 運算便利指令	FUN30 PID
<p>● PID 參數調整</p> <ul style="list-style-type: none"> <li>● 依下列原則適當調整 PID 參數以得到所要之程控反應：</li> <li>● 比例帶 (Pb) 調整越小，即增益越大，對輸出貢獻越大，可得到較快且靈敏之控制反應。但增益如過大，會造成振盪現象；儘量調高增益（但以不造成振盪為原則），以增快程序反應並減少穩態誤差。</li> <li>● 積分項可用來消除程控反應之穩態誤差。積分常數 (Ti) 調整越大，對輸出貢獻越大，當有穩態誤差時，可調高積分常數，以減少穩態誤差。 積分常數=0 時，積分項無作用。 如已知積分時間為 6 分鐘，則 <math>Ti=100/6=17</math>；如積分時間為 5 分鐘，則 <math>Ti=100/5=20</math>。</li> <li>● 微分項可用來讓程控反應較平順，不會造成過度超越。微分常數 (Td) 調整越大，對輸出貢獻越大，當有過度超越時，可調高微分常數，以減少超越量。微分項對程控反應相當靈敏，大部分之應用不必使用微分項，而將其設定為 0。微分常數=0 時，微分項無作用。 如已知微分時間為 1 分鐘，則 <math>Td=100</math>；如微分時間為 2 分鐘，則 <math>Td=200</math>。</li> </ul> <p><b>指令說明</b></p> <ul style="list-style-type: none"> <li>● 當控制選擇 "A/M" =0 時，代表手動控制模式，PID 運算不會執行。直接對輸出暫存器 (OR) 填值即可控制類比輸出量以控制程控反應。</li> <li>● 當控制選擇 "A/M" =1 時，代表自動控制模式，輸出暫存器 (OR) 之值由 PID 運算而來，經由類比輸出模組或其它界面控制程控負載以執行閉迴路程序控制。</li> <li>● 當平順轉移 "BUM" =1 時，由手動控制模式轉為自動控制模式時，控制輸出可平順銜接。</li> <li>● 當控制選擇 "A/M" =1 且運作方向 "D/R" =1 時，程序控制為順向 PID 控制；亦即誤差 (SP-PVn) 為正時，PID 運算結果之控制輸出量越大；誤差為負時，PID 運算結果之控制輸出量越小。</li> <li>● 當控制選擇 "A / M" =1 且運作方向 "D/R" =0 時，程序控制為反向 PID 控制；亦即誤差 (SP-PVn) 為正時，PID 運算結果之控制輸出量越小；誤差為負時，PID 運算結果之控制輸出量越大。</li> <li>● 當程控設定值或參數設定值錯誤時，PID 指令不會執行，設定錯誤指示 "ERR" =1。</li> <li>● 當程控值 <math>\geq</math> 上限設定值時，上限警告指示 "HA" =1。</li> <li>● 當程控值 <math>\leq</math> 下限設定值時，下限警告指示 "LA" =1。</li> </ul>		

FUN30 PID	PID 運算便利指令	FUN30 PID
<ul style="list-style-type: none"> <li>● PID 運算間隔時間暫存器說明： <ul style="list-style-type: none"> <li>● Ts：PID 運算間隔時間設定值暫存器，單位為 0.01 秒。在自動控制模式下，每隔所設定之間隔時間即作一次 PID 運算。</li> </ul> </li>   <li>● 程控設定值暫存器說明： <ul style="list-style-type: none"> <li>●SR+0：經轉換為工程單位後之程控值或稱實際讀值（Scaled Process Variable）；亦即 PID 指令會將原始類比輸入值轉換為我們所熟悉之物理量。</li> <li>●SR+1：設定值暫存器（Setpoint，簡稱 SP）；由使用者設定所期望之程控最終穩定值，其為工程單位。 正確之 SP 範圍為：<math>LER \leq SP \leq HER</math></li> <li>●SR+2：上限警告設定值暫存器（High Alarm Limit，簡稱 HAL）；由使用者設定當程控值<math>\geq</math>上限設定值時，上限警告指示“HA”=1；其為工程單位。 正確之 HAL 範圍為：<math>LER \leq LAL &lt; HAL \leq HER</math></li> <li>●SR+3：下限警告設定值暫存器（Low Alarm Limit，簡稱 LAL）；由使用者設定當程控值<math>\leq</math>下限設定值時，下限警告指示“LA”=1；其為工程單位。 正確之 LAL 範圍為：<math>LER \leq LAL &lt; HAL \leq HER</math></li> <li>●SR+4：程控最大工程值暫存器（High Engineering Range，簡稱 HER）；由使用者設定當原始類比輸入值為最大時所對應之實際最大物理量。 正確之 HER 範圍為：<math>-9999 &lt; HER \leq 9999</math></li> <li>●SR+5：程控最小工程值暫存器（Low Engineering Range，簡稱 LER）；由使用者設定當原始類比輸入值為最小時所對應之實際最小物理量。 正確之 LER 範圍為：<math>-9999 \leq LER \leq LAL &lt; HAL \leq HER</math></li> <li>●SR+6：原始類比輸入值暫存器（Raw Analog Measurement，簡稱 RAM）；使用者將類比輸入暫存器（R3840~R3903）所量測到之程控變數值（-2048 ~ 2047）加上 2048 後存放至此。 正確之 RAM 範圍為：<math>0 \leq RAM \leq 4095</math></li> <li>●SR+7：原始類比輸入偏差值暫存器（Offset of Process Variable，簡稱 OPV）。使用者所使用感知器之類比量測界面如與類比輸入模組所提供界面完全吻合，則 OPV 之值必須設定為 0；如使用者使用 4~20mA 等有偏移量之感知器，而類比輸入模組無提供偏移模式而僅提供 0~20mA 界面時，可設定 OPV 之值為 819（<math>4095 \times 4 / 20 = 819</math>）。 正確之 OPV 範圍為：<math>0 \leq OPV &lt; 4095</math></li> </ul> </li>   <li>●當上述程控設定值不在正確設定範圍內時，PID 指令不會執行，設定錯誤指示“ERR”=1。</li>   <li>● PID 輸出暫存器使用說明： <ul style="list-style-type: none"> <li>● OR：手動控制模式時，PID 運算不會執行。直接設定輸出暫存器（OR）之值即可控制類比輸出量以控制程控反應。 自動控制模式時，輸出暫存器（OR）之值是根據設定值與實際值之誤差經由 PID 運算而得出之控制輸出量。 正確之 OR 之範圍為：<math>0 \leq OR \leq 4095</math></li> </ul> </li> </ul>		

FUN30 PID	PID 運算便利指令	FUN30 PID
<p>● 參數設定值暫存器說明：</p> <ul style="list-style-type: none"> <li>● PR+0：比例帶設定值暫存器（Proportional Band，簡稱 Pb）；比例帶調整越小，即增益越大，對輸出貢獻越大，可得到較快且靈敏之控制反應。 正確之比例帶範圍為：<math>2 \leq Pb \leq 5000</math>，單位為 0.1% <math>Kc</math>（增益）=1000/ Pb</li> <li>● PR+1：積分常數（積分時間之倒數）設定值暫存器（Integral Constant，簡稱 Ti）；積分項可用來消除程控反應之穩態誤差。積分常數調整越大，對輸出貢獻越大。當有穩態誤差時，可調高積分常數，以減少穩態誤差。 積分常數=0 時，積分項無作用。 正確之積分常數範圍為：<math>0 \leq Ti \leq 9999</math>，0.00~99.99 Repeats/Minute 如已知積分時間為 6 分鐘，則 <math>Ti=100/6=17</math>； 如積分時間為 5 分鐘，則 <math>Ti=100/5=20</math>。</li> <li>● PR+2：微分常數設定值暫存器（Rate Time Constant，簡稱 Td）；微分項可用來讓程控反應較平順，不會造成過度超越。微分常數調整越大，對輸出貢獻越大，當有過度超越時，可調高微分常數，以減少超越量。 微分項對程控反應相當靈敏，大部分之應用不必使用微分項，而將其設定為 0。 微分常數=0 時，微分項無作用。 正確之微分常數範圍為：<math>0 \leq Td \leq 9999</math>，0.00~99.99 Minutes 如已知微分時間為 1 分鐘，則 <math>Td=100</math>；如微分時間為 2 分鐘，則 <math>Td=200</math></li> <li>● PR+3：輸出偏置設定暫存器（Bias）；當 PI 或 PID 程控時，如需加上偏置輸出時可設定此值。大部分之應用不必使用偏置輸出，而將其設定為 0。當僅為比例控制時（<math>Ti=0</math>，<math>Td=0</math>），為了消除穩態誤差，有需要設定偏置值。 正確之偏置值範圍為：<math>0 \leq Bias \leq 4095</math></li> <li>● PR+4：輸出上限設定暫存器（High Wind_up Limt，簡稱 HWL）；使用者可根據程控反應對輸出量作限制，使輸出量不會過大。大部分之應用皆不必作輸出限制，而將其設定為 4095。 正確之上限設定範圍為：<math>0 \leq HWL \leq 4095</math></li> <li>● PR+5：輸出下限設定暫存器（Low Wind_up Limt，簡稱 LWL）；使用者可根據程控反應對輸出量作限制，使輸出量不會過小。大部分之應用皆不必作輸出限制，而將其設定為 0。 正確之下限設定範圍為：<math>0 \leq LWL \leq 4095</math></li> <li>● PR+6：PID 方法選擇暫存器（PID Method）； =0，標準 PID 方法； =1，最小超越法（Minimum Overshoot Method）； 大部分之應用為 PI（<math>Td=0</math>）控制，建議選擇 Method 0。當使用 PID 控制時，如控制結果不是很穩定時，可試 Method 1 方法。</li> <li>● 當上述參數設定值不在正確設定範圍內時，PID 指令不會執行，設定錯誤指示“ERR”=1。</li> </ul>		

FUN30 PID	PID 運算便利指令	FUN30 PID
<p>● 工作暫存器說明：</p> <ul style="list-style-type: none"> <li>● WR+0：B0=0，手動控制模式 =1，自動控制模式 B1=1，完成 PID 運算，ON 一個掃描時間 B2=1，平順轉移 B4=1，“ERR” =1 B5=1，“HA” =1 B6=1，“LA” =1</li> <li>● WR+1~WR+4：系統內部使用。</li> </ul>		
<div style="border: 1px solid black; padding: 2px; display: inline-block;">PID 範例程式</div>		
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>● 將類比輸入暫存器所量測到之值加上 2048 後存放至 R1006，當作 PID 運算所需之原始類比輸入值。</p> <p>X0</p> <p>● X0=0，手動控制模式。 =1，自動控制模式。</p> <p>● R1010 為 PID 運算結果暫存器。 ● 將 PID 之運算結果減去 2048 後，存放至類比輸出暫存器，以控制類比輸出量。</p> </div> <div style="width: 50%; border: 1px solid black; padding: 5px;"> <div style="border: 1px solid black; padding: 2px; margin-bottom: 10px;"> <p>11.(+)</p> <p>Sa : R3840</p> <p>Sb : 2084</p> <p>D : R1006</p> </div> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%;"> <p>A/M</p> <p>BUM</p> <p>D/R</p> </div> <div style="width: 45%; border: 1px solid black; padding: 2px;"> <p>30.PID</p> <p>Ts : R 999</p> <p>SR : R1000</p> <p>OR : R1010</p> <p>PR : R1020</p> <p>WR : R1030</p> </div> <div style="width: 10%; text-align: right;"> <p>ERR—( ) Y0</p> <p>HA —( ) Y1</p> <p>LA —( ) Y2</p> </div> </div> </div> <div style="border: 1px solid black; padding: 2px; margin-top: 10px;"> <p>12.(-)</p> <p>Sa : R1010</p> <p>Sb : 2048</p> <p>D : R3904</p> </div> </div>		

