

51 单片机 PID 算法程序(三)增量式 PID 控制算法

当执行机构需要的不是控制量的绝对值，而是控制量的增量（例如去驱动步进电动机）时，需要用 PID 的“增量算法”。

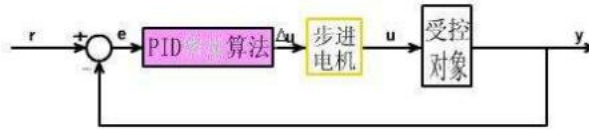


图3-2(b) 增量式PID控制算法的简化示意图

增量式 PID 控制算法可以通过 (2-4) 式推导出。由(2-4)可以得到控制器的第 k-1 个采样时刻的输出值为：

$$u_{k-1} = K_p \left[e_{k-1} + \frac{T}{T_i} \sum_{j=0}^{k-1} e_j + T_d \frac{e_{k-1} - e_{k-2}}{T} \right] \quad (2-5)$$

将 (2-4) 与 (2-5) 相减并整理，就可以得到增量式 PID 控制算法公式为：

$$\begin{aligned} \Delta u_k = u_k - u_{k-1} &= K_p \left[e_k - e_{k-1} + \frac{T}{T_i} e_k + T_d \frac{e_k - 2e_{k-1} + e_{k-2}}{T} \right] \\ &= K_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T} \right) e_k - K_p \left(1 + \frac{2T_d}{T} \right) e_{k-1} + K_p \frac{T_d}{T} e_{k-2} \\ &= A e_k - B e_{k-1} + C e_{k-2} \end{aligned} \quad (2-6)$$

其中

$$A = K_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T} \right)$$

$$B = K_p \left(1 + \frac{2T_d}{T} \right)$$

$$C = K_p \frac{T_d}{T}$$

由 (2-6) 可以看出，如果计算机控制系统采用恒定的采样周期 T，一旦确定 A、B、C，只要使用前后三次测量的偏差值，就可以由 (2-6) 求出控制量。

增量式 PID 控制算法与位置式 PID 算法 (2-4) 相比，计算量小得多，因此在实际中得到广泛的应用。位置式 PID 控制算法也可以通过增量式控制算法推出递推计算公式：

$$u_k = u_{k-1} + \Delta u, \quad (2-7)$$

(2-7) 就是目前在计算机控制中广泛应用的数字递推 PID 控制算法。

增量式 PID 控制算法 C51 程序

```

/*=====
PID Function
The PID (比例、积分、微分) function is used in mainly
control applications. PIDCalc performs one iteration of the PID
algorithm.
While the PID function works, main is just a dummy program showing
a typical usage.
=====*/
typedef struct PID
{
int SetPoint; //设定目标 Desired Value
long SumError; //误差累计

```

