

将普通硬件 RC 低通滤波器的微分方程用差分方程来表求，变可以采用软件算法来模拟硬件滤波的功能，经推导，低通滤波算法如下：

$$Y_n = a * X_n + (1-a) * Y_{n-1}$$

式中 X_n ——本次采样值

Y_{n-1} ——上次的滤波输出值；

， a ——滤波系数，其值通常远小于 1；

Y_n ——本次滤波的输出值。

由上式可以看出，本次滤波的输出值主要取决于上次滤波的输出值（注意不是上次的采样值，这和加权平均滤波是有本质区别的），本次采样值对滤波输出的贡献是比较小的，但多少有些修正作用，这种算法便模拟了具有较大惯性的低通滤波器功能。滤波算法的截止频率可用以下式计算：

$$f_L = a / (2\pi t) \quad \pi \text{ 为圆周率 } 3.14\dots$$

式中 a ——滤波系数；

， t ——采样间隔时间；

例如：当 $t=0.5s$ （即每秒 2 次）， $a=1/32$ 时；

$$f_L = (1/32) / (2 * 3.14 * 0.5) = 0.01\text{Hz}$$

当目标参数为变化很慢的物理量时，这是很有效的。另外一方面，它不能滤除高于 $1/2$ 采样频率的干扰信号，本例中采样频率为 2Hz，故对 1Hz 以上的干扰信号应采用其他方式滤除，

低通滤波算法程序于加权平均滤波相似，但加权系数只有两个： a 和 $1-a$ 。为计算方便， a 取一整数， $1-a$ 用 $256-a$ ，来代替，计算结果舍去最低字节即可，因为只有两项， a 和 $1-a$ ，均以立即数的形式编入程序中，不另外设表格。虽然采样值为单元字节（8 位 A/D）。为保证运算精度，滤波输出值用双字节表示，其中一个字节整数，一字节小数，否则有可能因为每次舍去尾数而使输出不会变化。

设 Y_{n-1} 存放在 30H（整数）和 31H（小数）两单元中， Y_n 存放在 32H（整数）和 33H（小数）中。