

第23章 检测算法

- 23.1 常用数字滤波方法
- 23.2 克服系统误差的软件算法
- 23.3 量程自动切换及标度变换

23. 检测算法

23.1 常用数字滤波方法

23.1.1 数字滤波技术特点概述

优点:

- (1) 不需增加硬件，只是一个计算程序，可靠性高，尤其是可对频率很高或很低的信号滤波。
- (2) 用软件算法实现，可以使多个输入通道共享一个软件“滤波器”，降低硬件成本。
- (3) 改变软件滤波器程序和参数，即可改变滤波特性，对于抑制低频脉冲干扰、随机噪声特别有效。

主要不足:

需要计算时间，可能影响实时性。

23.1 常用数字滤波方法

23.1.1 数字滤波方法

1) 程序判断滤波

$$\Delta y_n = |y_n - \bar{y}_{n-1}| \begin{cases} \leq a & \bar{y}_n = y_n \\ > a & \bar{y}_n = \bar{y}_{n-1} \text{ 或 } \bar{y}_n = 2\bar{y}_{n-1} - \bar{y}_{n-2} \end{cases}$$

α : 相邻两个采样值之差的最大的可能变化范围。

程序判断滤波又称限幅滤波，很容易用程序判断方法实现。

关键：确定 α 的值。

因过程的动态特性决定其输出参数的变化速度，故可根据检测实践经验或按参数可能的最大变化速度 V_{\max} 及采样周期 T 来决定 α 值，即： $\alpha = T \cdot V_{\max}$

23.1 常用数字滤波方法

23.1.1 数字滤波方法

2) 中(位)值滤波法

对某一被测参数连续采样 n 次(一般 $n=2m+1$), 把 n 次采样值按大小排序, 取中间值作为本次的有效采样值。

对温度、液位等缓慢变化的被测参数, 此法能收到良好滤波效果, 但对压力等快变参数一般不宜采用。

特点: 有效抑制脉冲(宽度 $< m$)干扰, 保持边沿或细节特征。

3) 算术平均滤波法

取连续几个采样值的算术均值 $y = \frac{1}{N} \sum_{i=1}^N y_i$ 作为滤波输出值。

此方法适用于一般有随机干扰的信号滤波, 对信号的平滑程度完全取决于 N 。 N 大, 平滑度高, 但灵敏度低; N 小, 平滑度低, 但灵敏度高。

23.1 常用数字滤波方法

23.1.1 数字滤波方法

4) 递推平均滤波法

把 N 个测量数据看成一个队列，前次所测数据作列尾，将此数为列尾数的前 N 个测量数据的算术平均值 $y_n = \frac{1}{N} \sum_{i=0}^{N-1} y_{n-i}$ 作为当前滤波输出值。

特点： 只需一次测量，就能得到当前的滤波输出值

5) 加权递推平均滤波法

$$y_n = \frac{1}{N} \sum_{i=0}^{N-1} C_i y_{n-i} \quad \text{其中: } \begin{cases} C_0 + C_1 + \dots + C_{N-1} = 1 \\ C_0 > C_1 > \dots > C_{N-1} > 0 \end{cases}$$

权值确定方法： 设 τ 为对象的纯滞后时间，且

$$\delta = 1 + e^{-\tau} + e^{-2\tau} + \dots + e^{-(N-1)\tau}, \quad \text{则: } C_0 = \frac{1}{\delta}, C_1 = \frac{e^{-\tau}}{\delta}, \dots, C_{N-1} = \frac{e^{-(N-1)\tau}}{\delta}$$

23.1 常用数字滤波方法

23.1.1 数字滤波方法

4) 复合滤波法

若 $y_1 \leq y_2 \leq \dots \leq y_n$, 其中, $3 \leq n \leq 14$ (y_1 和 y_n 分别是采样值中的最小值和最大值), 则: $\bar{y}_n = (y_2 + y_3 + \dots + y_{n-1}) / (n - 2)$

5) 一阶惯性滤波法

以数字形式通过算法实现一阶惯性 (动态) RC 滤波, 能很好地克服模拟滤波器的缺点。适用于波动频繁的参数滤波, 但带来相位滞后 (取决于 a 值), 灵敏度低。算法如下:

$$y_n = (1 - a)y_n - ay_{n-1}$$

其中, $a = T_f / (T_f + T)$, T_f 为滤波时间常数, T 为采样周期。 a 由实验定, 只要使被测信号不产生明显纹波。

缺点: 不能滤除频率高于采样频率二分之一 (奈奎斯特频率) 的干扰信号。高于奈奎斯特频率的干扰信号, 需模拟滤波。

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

校正系统误差的关键是建立误差模型。多数情况下模型未知，只能通过测量数据建立反映测量值变化的近似模型。

1) 代数插值法

设有 $n+1$ 组离散点: $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, $x \in [a, b]$ 和未知函数 $f(x)$, 并有 $f(x_0)=y_0, f(x_1)=y_1, \dots, f(x_n)=y_n$, 找一个函数 $g(x)$, 使 $g(x)$ 在 $x_i(i=0, \dots, n)$ 处与 $f(x_i)$ 相等。满足此条件的函数 $g(x)$ 称为 $f(x)$ 的插值函数, x_i 称为插值节点。

用一个 n 阶代数多项式 $P_n(x)=a_nx^n+a_{n-1}x^{n-1}+\dots+a_1x+a_0$ 逼近 $f(x)$, 使 $P_n(x)$ 在节点 x_i 处满足 $P_n(x_i) = f(x_i) = y_i \quad i=0, 2, \dots, n$

$$\begin{cases} a_n x_0^n + a_{n-1} x_0^{n-1} + \dots + a_1 x_0 + a_0 = y_0 \\ a_n x_1^n + a_{n-1} x_1^{n-1} + \dots + a_1 x_1 + a_0 = y_1 \\ \vdots \\ a_n x_n^n + a_{n-1} x_n^{n-1} + \dots + a_1 x_n + a_0 = y_n \end{cases}$$

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

通常给出离散点总是多于求解插值方程所需的离散数，因此，用多项式插值法求解离散点的插值函数时，先必须根据所需的逼近精度决定多项式阶次，该次数与所要逼近的函数有关。一般常用线性插值和抛物线(二次)插值。

(1) 线性插值

在一组数据 (x_i, y_i) 中选两个代表性的点 (x_0, y_0) , (x_1, y_1) ，然后根据插值原理求出插值方程：

$$P_1 = \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1 = a_1 x + a_0$$

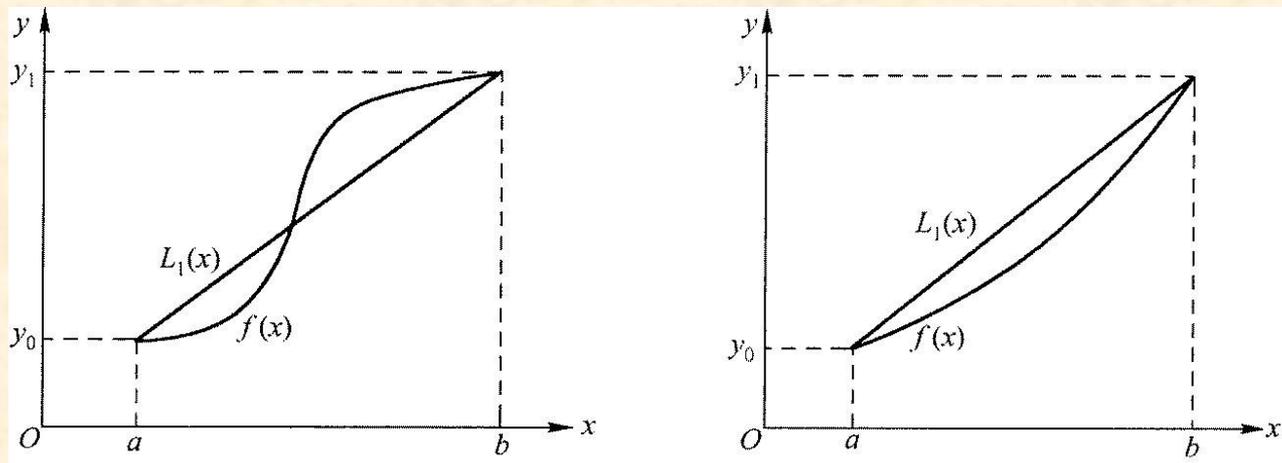
$$\text{其中： } a_1 = \frac{y_1 - y_0}{x_1 - x_0}, a_0 = y_0 - a_1 x_0$$

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

当 (x_0, y_0) 、 (x_1, y_1) 为非线性特性曲线 $f(x)$ 或数组的两端点时，线性插值是常用的直线方程校正法(端点连线法)。拟合误差小于允许拟合误差时，直线方程是理想的校正方程。实际测量中，每采样一个值，就用校正方程计算 $P_1(x)$ ，并把 $P_1(x)$ 当做被测量值的校正值。



非线性特性的直线方程校正图

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

以镍铬—镍铝热电偶(0~490°C)为例说明方法的应用。
要求：采用直线方程进行非线性校正，误差小于3°C。

求得直线校正方程 $P_1(x)=24.245x$ ，在两端点的拟合误差为0，而在 $x=11.38\text{mV}$ 时， $P_1(x)=275.91^\circ\text{C}$ ，误差：4.09°C，达最大值，240°C~360°C范围内拟合误差均大于3°C。

镍铬—镍铝热电偶分度表

温 度 /°C	0	10	20	30	40	50	60	70	80	90
	热 电 势/mV									
0	0.00	0.40	0.80	1.20	1.61	2.02	2.44	2.85	3.27	3.68
100	4.10	4.51	4.92	5.33	5.73	6.14	6.54	6.94	7.34	7.74
200	8.14	8.54	8.94	9.34	9.75	10.15	10.56	10.97	11.38	11.80
300	12.21	12.62	13.04	13.46	13.87	14.29	14.71	15.13	15.55	15.97
400	16.40	16.82	17.24	17.67	18.09	18.51	18.94	19.36	19.79	20.21

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

显然，对非线性程度严重或测量范围较宽的非线性特性，用一个直线校正方程进行校正，很难满足精度要求。

这时可用分段直线方程进行校正。分段后的每段非线性曲线用一个直线方程校正，即： $P_{1i}(x) = a_{1i}x + a_{0i} \quad i=1, 2, \dots$ 。

分段的折线的节点有等距与非等距两种取法。

(1) 等距节点分段直线校正

适用于非线性曲线曲率变化不大的场合。每段曲线都用一个直线方程代替。分段数 N 取决于非线性程度和系统精度要求。非线性越严重或系统精度要求越高， N 越大。为便于实时计算，常取 $N=2m$ ， $m=0, 1, \dots$ 。 $P_{1i}(x) = a_{1i}x + a_{0i}$ 中的 a_{1i} 和 a_{0i} 可离线求得。

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

(1) 等距节点分段直线校正

采用等分法，每段的拟合误差均不同，应保证最大的拟合误差不大于允许误差。把求得的 a_{1i} 和 a_{0i} 存入系统ROM中，实测时只要先用程序判断输入位于哪段折线，然后取出该段对应的 a_{1i} 和 a_{0i} 计算，可得到被测量的相应近似值。

对曲率变化大和切线斜率大的非线性特性用等距节点法校正，要使最大误差满足要求，分段数会很大，而误差分配却不均。同时分段数增加，使 a_{1i} 和 a_{0i} 的数目相应增加，占用内存较多。

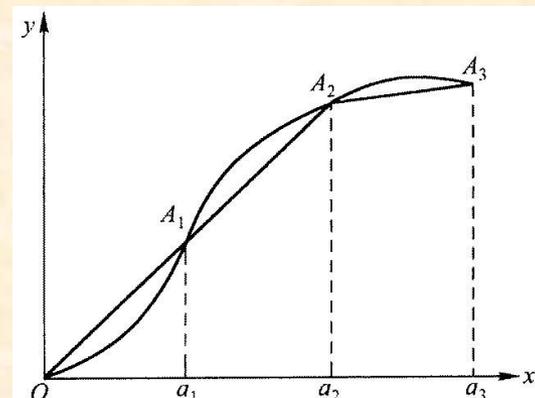
23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

(2) 非等距节点分段直线校正

在线性较好的部分节点间距大，反之小，从而使误差达到均匀分布。右图为采用不等分的三段折线校正。



非等距节点分段直线校正图

$$P_1(x) = \begin{cases} a_{11}x + a_{01} & 0 \leq x < a_1 \\ a_{12}x + a_{02} & a_1 \leq x < a_2 \\ a_{13}x + a_{03} & a_2 \leq x < a_3 \end{cases}$$

以前表数据为例，用等距节点分段直线校正法，从表中取三点：(0, 0)，(10.15, 250)及(20.21, 490)，用经过这三点的两个直线方程近似代替整个表格，可求得方程为：

$$P_1(x) = \begin{cases} 24.63x & 0 \leq x < 10.15 \\ 23.86x + 7.85 & 10.15 \leq x < 20.21 \end{cases}$$

可验证，用上式校正非线性后每点的误差均不大于 2°C 。 13

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

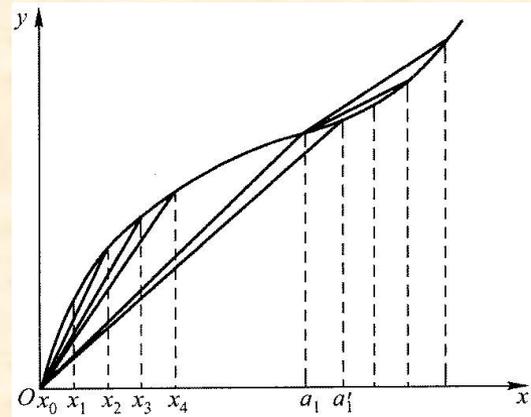
1) 代数插值法

(2) 非等距节点分段直线校正

某些情况下，因非线性特性不规则，在两端点间取的第三点可能不合理，导致误差非均匀分布。尤其是非线性严重时，用一段或两段直线拟合无法保证拟合精度，需通过增加分段或采用非等距节点分段直线校正法来满足要求。

对非线性不规则的情况，合理确定分段数和节点是关键。

一种根据允许误差确定分段数和节点的方法：设有图示非线性特性。从一个端点出发，先求出 $[x_0, x_2]$ 间的直线方程，并求出中间点 x_1 的拟合误差。若该误差小于允许误差，则再延伸一个点 x_3 ，求出 $[x_0, x_3]$ 直线方程，并求出中间点的最大拟合误差：



$$V_{\max} = \max |V(x_i)| \quad i=1, 2。$$

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

1) 代数插值法

(2)非等距节点分段直线校正

这样一直进行下去直到在区间 $[x_0, a'_1]$ 至少有一个点，在该点的拟合误差超出允许拟合误差，则以起始点到 a'_1 的前一点 a_1 为第一段，再从第一节点开始搜索第二段，...，直到结束。求出的方程系数 a_{1i} 和 a_{0i} 存入内存，以备实时计算用。

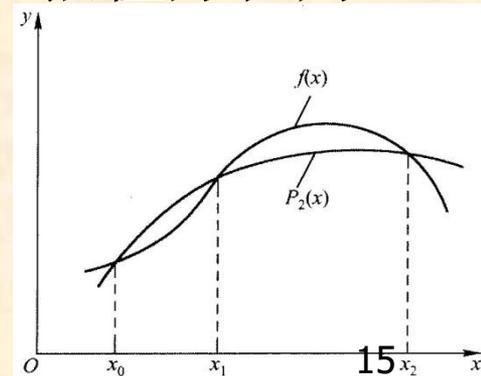
非等距法可最大程度减少校正段数，提高精度。

2) 抛物线插值(选讲)

在数据中选三点 (x_0, y_0) 、 (x_1, y_1) 、 (x_2, y_2) ，插值方程为：

$$P_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}y_2$$

其几何意义如图所示。选 $(0, 0)$ 、 $(10.15, 250)$ 和 $(20.21, 490)$ ，得： $P_2(x) = -0.038x^2 + 25.02x$ 。
可验证，方程校正后每点误差均小于 3°C 。



23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

2) 最小二乘法（选讲）

对含随机误差的实验数据的拟合，常用“误差平方和最小”标准来衡量逼近结果，使逼近模型较为符合实际关系，形式上简单。

设被逼近函数为 $f(x_i)$ ，逼近函数为 $g(x_i)$ ， x_i 为 x 上的离散点，逼近误差为： $V(x_i)=|f(x_i)-g(y_i)|$

记： $\varphi = \sum_{i=1}^n V^2(x_i)$

令 $\varphi \rightarrow \min$ ，即在最小二乘意义上使 $V(x)$ 最小化。为了使逼近函数简单，常选择 $g(x)$ 为多项式。

最小二乘拟合法有直线拟合和曲线拟合两种。当函数类型确定后，函数关系中的一些待定系数，常用最小二乘法来确定。

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

2) 最佳一致逼近法（选讲）

实际问题往往是要求 $g(x)$ 在整个测量区间的每一点上都很好的逼近 $f(x)$ ，插值法不能取得满意效果。对此，可采用最佳一致逼近法。

最佳一致逼近法是保证 $f(x)$ 与 $g(x)$ 之间的最大误差小于给定值，即使得不等式成立： $\max|g(x)-f(x)|\leq\varepsilon$ ；其中 $a\leq x\leq b$ ， a 、 b 为测试区间的端点。

线性最佳一致逼近：找一条直线 $P_1^*(x) = a_0 + a_1x$ ，使其与所有对应 x_i 的纵坐标 y_i 之差的绝对值与其他任意一直线相比， $\max|P_1^*(x)-f(x_i)|$ 为最小。式中 a_0 和 a_1 待定。

几何意义：作平行于弦AB并与 $f(x)$ 相切的直线，切点为C。过AC之中点D作AB的平行线 $P_1^*(x)$ ，则 P_1^* 即为 $f(x)$ 的线性最佳一致逼近直线方程。

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

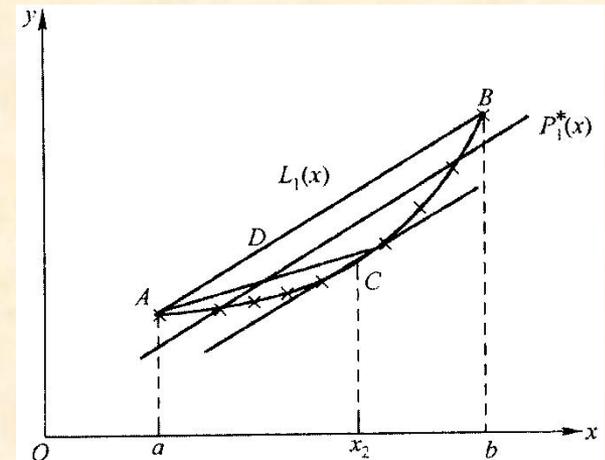
2) 最佳一致逼近法（选讲）

设被逼近函数 $f(x)$ 单调上凸或下凹，其线性最佳一致逼近方程为： $P_1^*(x) = a_0 + a_1x$ ；其中，

$$a_1 = \frac{f(b) - f(a)}{b - a}$$

$$a_0 = \frac{f(a) + f(x_2)}{2} - a_1 \frac{a + x_2}{2}$$

式中 x_2 是满足 $P_1^*(x_2) - f(x_2) = 0$ 的 x 值，即 x_2 是右图中切点C的横坐标。



最佳直线方法

3) 分段线性最佳一致逼近（选讲）

单个线性最佳一致逼近方程无法满足非线性校正精度要求时，可用分段线性最佳一致逼近法。其节点选择也有等距和不等距两种。

23.2 克服系统误差的软件算法

23.2.1 系统误差的模型校正法（非线性校正）

三种不同的离散数据直线逼近方法特点比较（选讲）：

- 线性最佳一致逼近能保证最大绝对值误差最小，但不保证误差平方和最小；最小二乘法保证误差平方和最小，但不能保证绝对误差最小。
- 一般情况下，对同一非线性特性函数 $f(x)$ ，若校正误差 ε 相同，则用最佳一致逼近校正比折线校正所分的段数要少得多；反之，若段数相同，则前者校正精度要高得多。
- 分段折线校正，在节点上的校正值是唯一的，节点两侧的校正值一般不会突变；用线性一致逼近时，在节点上校正不唯一，甚至误差的极性可能完全相反。

23.2 克服系统误差的软件算法

23.2.2 系统误差的标准数据校正法（查表法）

当合适的误差校正模型难建时，可用实验求得校正曲线，再把曲线上各校正点的数据以表格形式保存。实时测量时，通过查表求得修正后的测量结果。

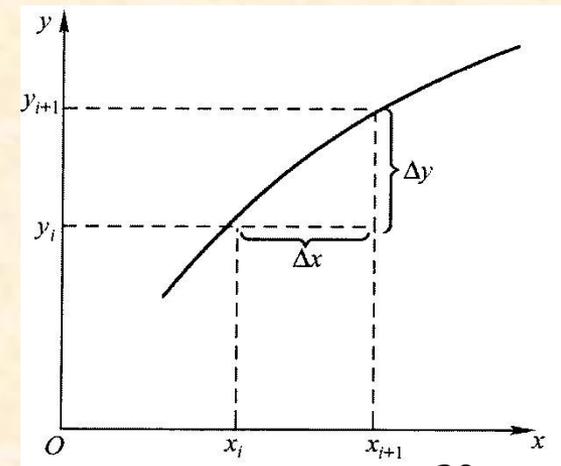
当实测值 y 介于校正点 y_i 和 y_{i+1} 之间时，若是直接查表，则只能按其最接近的 y_i 和 y_{i+1} 查，显然会引入一定的误差。

误差可作如下估计，设两校正点间的校正曲线为一直线段，其斜率 $S = \Delta x / \Delta y$ （校正时 y 是自变量， x 是函数值）。设最大斜率为 S_m ，由图可见，可能的最大误差为： $\Delta x_m = S_m \Delta y$ 。

考虑取双向误差的绝对值可减半，则最大误差为： $\pm \Delta x_m = \pm S_m \Delta y / 2$ 。

设 y 的量程为 Y ，等间隔取 N 个校正点，则：

$$y_{i+1} - y_i = \Delta y = Y/N; \quad \Delta x_m = S_m \cdot Y/2N.$$



23.2 克服系统误差的软件算法

23.2.3 系统零位误差和增益误差的校正方法

传感器和测量电路不可避免地存在温漂和时漂，给整个系统引入零位误差和增益误差。这类误差均属系统误差。

1) 零位误差的校正方法

■ 在每个测量周期中或中断正常的测量过程中，使输入接地(输入为零)，此时包括传感器在内的整个测量输入通道的输出即为零位输出(一般该值不为零) N_0 ；再把输入接基准电压 V_R 测得数据 N_R ，并将 N_0 和 N_R 存于内存。然后输入接 V_x ，测得 N_x ，则测量结果可用下式计算：

$$V_x = V_R (N_x - N_0) / (N_R - N_0)$$

即在正常测量过程中，每次测量后都从采样值中减去原先存入的零位输出值，实现零位校正。对温度、流量、压力等的工业测量中，很难提供零输入状态，零位误差难补偿。

23.2 克服系统误差的软件算法

23.2.3 系统零位误差和增益误差的校正方法

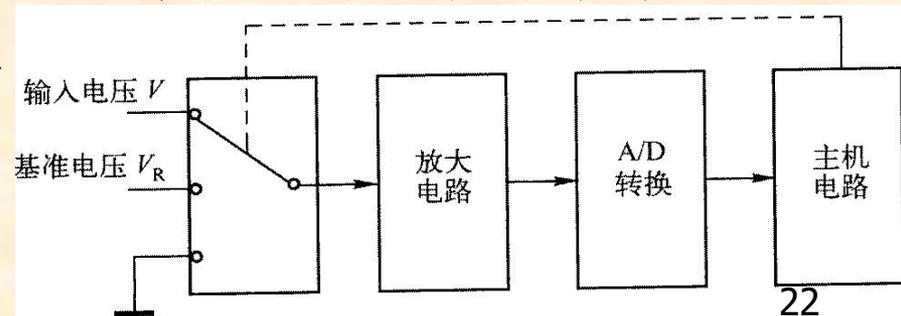
2) 增益误差的校正方法

基本思想：系统开始工作后或每隔一定时间测一次基准参数，然后建立误差校正模型，确定并存储校正模型参数。正式测量时，根据测量结果和校正模型求取校正值修正误差。

(1) 全自动校正

校正时，先将开关接地，测出这时的输入 x_0 。然后把开关接 V_R ，测出输入 x_1 ，存放 x_0, x_1 ，得校正方程： $y = a_1x + a_0$ ，式中， $a_1 = V_R / (x_1 - x_0)$ ， $a_0 = V_R x_0 / (x_0 - x_1)$ 。

此方法测得的信号与放大器的漂移和增益变化无关，可大大提高测量精度，降低对电路器件的要求。



23.2 克服系统误差的软件算法

23.2.3 系统零位误差和增益误差的校正方法

2) 增益误差的校正方法

(2) 人工自动校正

由人工在需要时接入标准参数进行校正测量，存放所测数据，供以后使用。一般人工自动校正只测一个标准信号 y_R ，零位信号的补偿由数字调零完成。设数字调零后测得的数据分别为 x_R (接标准输入 y_R 时)和 x (接被测输入 y 时)，可计算出：

$$y = y_R x / x_R$$

人工自动校正特别适于传感器特性随时间变化的场合。如常用的湿敏电容等湿度传感器，其特性随时间而变化。

全自动校正只适用于基准参数是电信号的场合，且不能校正由传感器引入的误差。

23.2 克服系统误差的软件算法

23.2.4 传感器温度误差的校正方法

只要能建立较精确的温度误差数学模型，就可实现完善的补偿。可行方法是在传感器内靠近敏感元件处安装一测温元件，感受传感器的工作环境温度及变化($\Delta\theta$)。

对某些传感器，可用较简单的温度误差校正数学模型

$$y_C = y(1 + a_0\Delta\theta) + a_1\Delta\theta$$

式中： y 是未经温度误差校正的测量值； y_C 是经校正后的测量值； $\Delta\theta$ 是实验工作环境温度与标准温度之差； a_0 、 a_1 是温度变化系数， a_1 用于补偿零位漂移， a_0 用于补偿传感器灵敏度的温度变化（温漂）。

当环境温度变化较大，且补偿精度要求很高时，需用更复杂的公式修正： $y_C = y(1 + a_0\Delta\theta + a_1\Delta\theta^2) + a_2\Delta\theta + a_3\Delta\theta^2$ 。

式中 a_0 , a_1 用于补偿灵敏度变化； a_2 , a_3 用于补偿零位漂移。

23.3 量程自动切换及标度变换

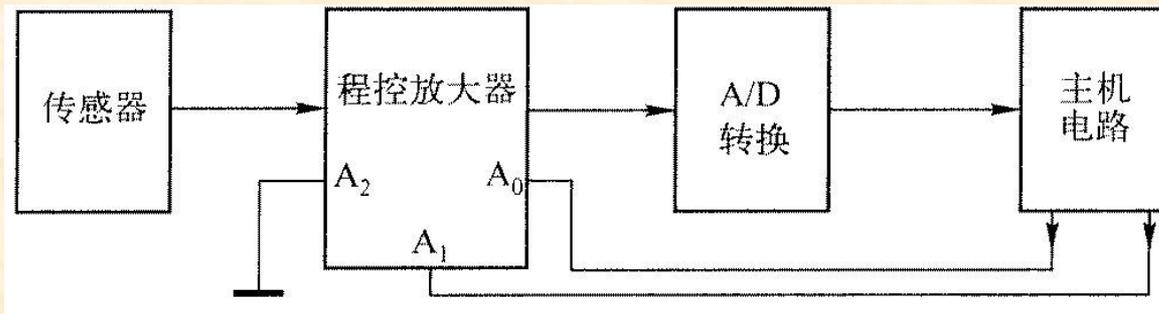
23.3.1 量程自动切换

如果传感器和显示器的分辨率一定，而系统(仪表)的测量范围很宽，为提高测量精度，系统应自动切换量程。

自动切换量程的两途径：程控放大、不同量程传感器

1) 程控放大途径

通过控制增益，对小幅值信号用大增益，对大幅值信号用小增益，使A/D转换器信号满量程达到均一化，程控放大器的反馈回路中含精密梯形电阻网络或权电阻网络，使增益可按二进制或十进制规律进行控制。



程控放大器量程切换原理图

23.3 量程自动切换及标度变换

23.3.1 量程自动切换

1) 程控放大途径

设有一压力传感器，量程： $0\sim 1\text{MPa}$ ，精度： $\pm 0.1\%$ 。若把测量范围压缩到 $0\sim 0.1\text{MPa}$ ，其相对精度仍可达 $\pm 0.2\%$ 。在此情况下，用程控放大可充分发挥传感器的性能。

选3位半的ADC，量程分为 $0\sim 1\text{MPa}$ 和 $0\sim 0.1\text{MPa}$ 。小量程时，传感器输出小，通过程控放大器的增益补偿，使单位数字量所代表的压力减小，提高数字计算的分辨率。

使 $0\sim 1\text{MPa}$ 量程的程控增益为1，控制线 $A_2A_1A_0=000\text{B}$ ，当被测压力最大时，A/D输出为1999。此量程内，一旦A/D输出小于200，经软件判断后自动转入小量程 $0\sim 0.1\text{MPa}$ ，并使放大器增益提高到8，即令 $A_2A_1A_0=011\text{B}$ 。当小量程档内A/D的输出在 $[200, 1600]$ 时，经软件判断后自动转入大量程，增益恢复为1。

23.3 量程自动切换及标度变换

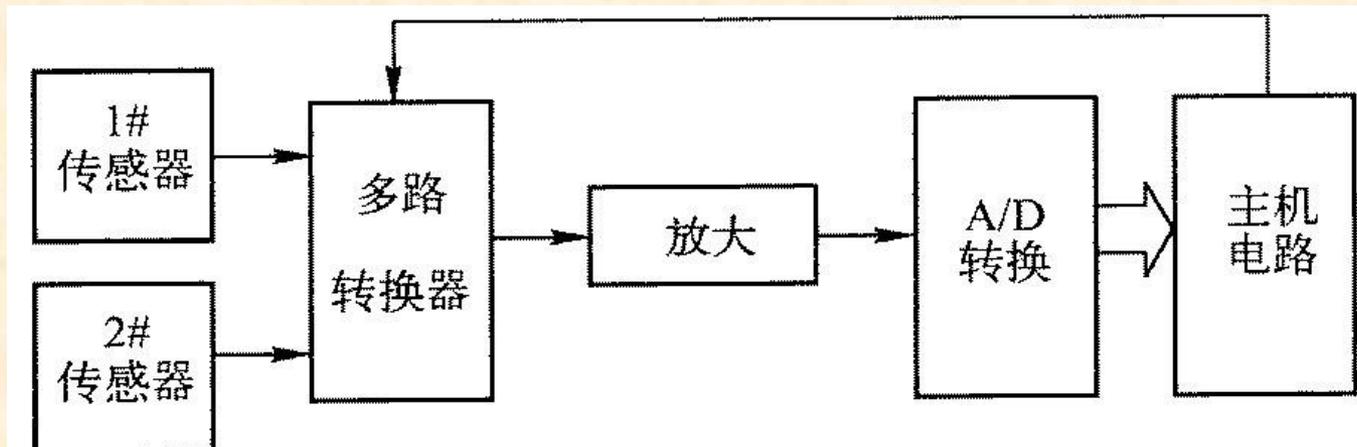
23.3.1 量程自动切换

2) 采用不同量程的传感器

配置不同量程的传感器，由微机通过多路转换器切换。

例：1#传感器的最大测量范围为 M_1 ，2#为 M_2 ，且 $M_1 > M_2$ ，设它们的满量程输出电压相同。

启动时，总是1#传感器先接入工作，2#处于过载保护，待软件判别确认量程后，再置标志位，选取 M_1 或 M_2 。



23.3 量程自动切换及标度变换

23.3.2 标度变换

实用中，测出被测模拟量并转换为数字量后，常需转换成操作员熟悉的工程量。因被测对象的各种数据的量纲与A/D转换的输入值不同，这些参数经传感器和A/D转换后得到一系列数码，这些数码并不等于原来带量纲的参数值，仅对应于参数大小，必须将其转换成带量纲的数值后才能运算、显示和输出，这种转换是**工程量变换**，又称**标度变换**。

可分为线性系统标度变换和非线性参数标度变换。

1) 线性系统

标度变换公式为：
$$A_x = A_0 + (A_m - A_0) \frac{N_x - N_0}{N_m - N_0}$$

式中： A_0 为测量范围下限； A_m 为测量范围上限； N_m 是 A_m 对应的数字； N_0 是 A_0 对应的数字； N_x 是被测量 x 对应的数字。

23.3 量程自动切换及标度变换

23.3.2 标度变换

1) 线性系统

例：已知某热处理炉温度测量系统的量程为200~800°C，在某时刻计算机采样并经数字滤波后的数字量(8位)为0CDH，此时的温度值是多少？

解：已知 $A_0=200^\circ\text{C}$ ， $A_m=800^\circ\text{C}$ ， $N_0=0$ ， $N_m=0FFH=255$ ，

$N_x=0CDH=205$ ，此时温度：

$$A_x = N_x (A_m - A_0) / N_m + A_0 = 205 \times (800 - 200) / 255 + 200 = 628^\circ\text{C}$$

23.3 量程自动切换及标度变换

23.3.2 标度变换

2) 非线性参数的标度变换

被测量为非线性刻度时，变换式应根据具体问题分析，先求它对应的标度变换公式，再进行程序设计，或在测量范围内实测一些典型数据来制作数据表，再用查表程序实现标度变换，对中间数据可用插值法计算。

例：在流量测量中，流量和压差的公式为： $Q = K \sqrt{\Delta P}$

式中： Q 为流量； ΔP 为节流装置前后的差压； K 是刻度系数，与流体的性质及节流装置的尺寸形状有关。

根据流量与压差的关系式，测流量时的标度变换式为：

$$Q_x = (Q_m - Q_0) \sqrt{\frac{N_x - N_0}{N_m - N_0}} + Q_0$$

式中， Q_x 为被测流量值； Q_m 和 Q_0 为流量计的上、下限； N_x 为差压变送器所测差压值(数字量)； N_m 和 N_0 分别为差压变送器上、下限对应的数字量。

23.3 量程自动切换及标度变换

23.3.2 标度变换

2) 非线性参数的标度变换

对流量计，一般下限皆为0，即 $Q_0=0$ ，上式可简化为：

$$Q_x = Q_m \frac{N_x - N_0}{N_m - N_0}$$

若取流量表下限对应数字量 $N_0=0$ ，上式可进一步简化。

与线性刻度标度变换公式一样，由于 Q_m ， Q_0 ， N_m ， N_0 都是常数，所以公式可分别记作：

$$Q_{x1} = K_1 \sqrt{N_x - N_0} + Q_0, \text{其中 } K_1 = \frac{Q_m - Q_0}{\sqrt{N_m - N_0}}$$

$$Q_{x2} = K_2 \sqrt{N_x - N_0}, \text{其中 } K_2 = \frac{Q_m}{\sqrt{N_m - N_0}}$$

$$Q_{x3} = K_3 \sqrt{N_x}, \text{其中 } K_3 = \frac{Q_m}{\sqrt{N_m}}$$

左边三式为各种不同条件下的流量标度变换公式，同样可设计出流量标度变换程序。