

单片机系统中应用的几种数字滤波方式

杨 剑 刘光斌

(第二炮兵工程学院 西安 710025)

摘 要:在自动化测控系统中,数字滤波在对采样信号的处理上占据着重要的作用。针对以单片机为代表,对实时性和存储量有特殊要求的嵌入式系统,介绍了几种实用数字滤波方式,并给出实现相应滤波算法的原程序。

关键词:单片机 嵌入式系统 数字滤波 程序

Several Ways of Digital Filtering Application in MCU System

YANG Jian, LIU Guangbin

(The Second Artillery Engineering College, Xi'an, 710025, China)

Abstract: Digital filtering has important effect in processing sampled signals in auto-measure and auto-control systems. Aiming at MCU system, as the representation of embedded system that have some special demands of real time and memory capacity, some applied ways of digital filtering have been introduced in this paper, including the programs of realizing mathematics of those filtering.

Keywords: MCU, Embedded system, Digital filtering, Program

数字滤波是自动测控系统中,对测量信号进行采样后处理中的关键环节,是非常重要的和必要的。而由于单片机速度和存储容量都很有限,实际应用中由于实时性和存储量的限制,在普通单片机上要实现复杂的数字滤波是不太可能和实际的。因而,本文针对单片机等系统,讨论了一些数字滤波方式。所讨论的这几种数字滤波方式,有算法简单、存储空间消耗小和滤波效果好的特点,在各种算法后给出了具体实现的程序。

1 数字滤波的实现

1.1 均值滤波

均值滤波,顾名思义就是通过对信号多次采样再求其平均值。这样,随着采样次数的增加可提高信号的分辨率和降低干扰量的影响。

均值滤波的公式为

$$Y = \frac{1}{n} \sum_{i=1}^n X_i$$

式中: X_i 为第 n 次采样时的滤波器输入; n 为采样次数; Y 为取均值后滤波器输出。

在均值滤波中,采样次数和运算量、存储空间是矛盾的。在一定范围内,采样次数 n 越大,均值滤波的效果越好;但 n 越大,运算量和存储空间的开销也越大。因此,我们要综合考虑三者的关系,选择合适的 n 值。

均值滤波的实现程序为

```
# define N? //设定采样次数 n 的大小
int x[ N], y; //采样数组
long int sum;
int i;
.....
for( i= 0; i< N; i+ + )
x[ i]= cyzc( ); //返回采样值
sum= x[ 0];
for( i= 1; i< N; i+ + )
sum+ = x[ i];
y= sum/ N;
.....
```

1.2 限幅滤波

限幅滤波是克服输入中窜入尖脉冲干扰。其基本思想是将获得的采样值与参考值进行比较,根据经验设定允许的最大偏差。如果采样值和参考值的差值超过了允许的最大偏差,则认为本次采样值中窜入了干扰,应重新采样。

其软件实现如下:

```
# define CANKAO ? // 设定参考值
# define CHAZHI ? // 设定偏差范围
int k;
```

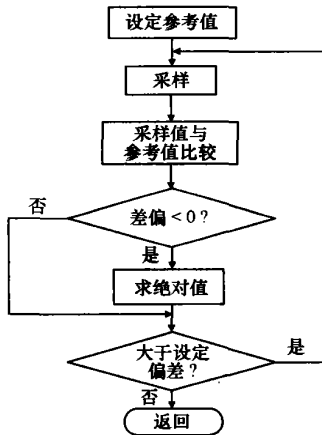


图1 限幅滤波流程图

```

.....
CY:
x= cyzc( );      // 获得采样值
k= x- CANKAO; // 作差值
if(k< 0)         // 判断差值是否 小于 0
k= - k;         // 小于 0 则取其绝对值
if(k> CHAZHI) // 如果大于偏差范围,则重采样
goto CY;
.....

```

1.3 低通滤波

一阶低通数字滤波器是用软件实现硬件 RC 滤波器, 以抑制干扰信号。一阶低通数字滤波公式为

$$Y_n = qX_n + (1-q)Y_{n-1}$$

式中: q 为数字滤波器的时间常数, 实际取值取决于滤波时间常数和采样周期; X_n 为第 n 次采样时的滤波器输入; Y_n 为第 n 次采样时滤波器输出。

若采样间隔 Δt 足够小, 则滤波器的截止频率为:

$$f_c = \frac{q}{2\pi \Delta t}$$

其算法流程如图2所示。

一阶低通数字滤波器在程序上的实现比较简单:

```

# define a 0.0565; // 若采样周期  $\Delta t$  为 300 us, 则
截止频率  $f_c = 30$  Hz

```

$$y(n) = a \cdot x + (1-a) \cdot y(n-1);$$

1.4 占优滤波

采样信号中常伴随着干扰信号的存在, 干扰信号的类型多样, 产生因素复杂。这里介绍一种占优滤波的方法适用于处理线性、连续变化的采样信号, 而不需要考虑干扰信号的类型, 可用于剔除采样信号中的无用信息。

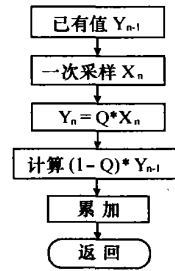


图2 一阶低通滤波流程图

型, 可用于剔除采样信号中的无用信息。

主要思想是: 设计一个长度为 n (n 可根据具体需要设定大小) 的基准向量 (初值可为 0), 基准向量内装载 n 次测量的值。在 $n+1$ 次测量后, 将测量值与基准向量进行比较, 取这向量中重复次数最多的值作为 $n+1$ 次测量的真值。

具体而言, 我们用长度为 n 的一维数组 $a(n)$ 作为基准向量:

$$a = \{a(0), a(1), \dots, a(n-1)\}$$

在进行 $n+1$ 次测量后得到测量值 q , 将基准向量内的值平移一位:

$$a(0) = a(1), a(1) = a(2), \dots,$$

$$a(n-2) = a(n-1)$$

将测量值 q 存到向量的最后一位:

$$a(n-1) = q$$

统计向量中各值重复次数为 $n(0), n(1), \dots, n(n-1)$:

$$n(k) = \max(n(0), n(1), \dots, n(n-1))$$

取向量中重复次数最大的值作为 $n+1$ 的测量真值:

$$q = a(k)$$

实现上述滤波方法的程序如下:

```

# define N 10 // 设定 n 的长度
int st[N]; // 定义基准向量
int szlb(int q) // q 为测量值
{
    int i, j, k, kk;
    for(i=0; i<N-1; i++)
        st[i] = st[i+1];
    st[N-1] = q;
    for(j=0; j<N-2; j++)
        { k = st[j];
        for(i=j+1; i<N-1; i++)
            { if(k == st[i])
                s[j] += 1;
            }
        }
    k = s[0];
}

```

```

kk= 0;
for(j= 1;j< N- 1;j+ + )
{ if( k< s[ j] )
{ k= s[j];
kk= j;}}
return st[kk]; // 返回经过处理后得到的测量真值
}

```

1.5 替代滤波

在测量信号中,干扰量的偶尔串进使得采样后的数字量中存在一些奇异值。替代滤波技术能够消除在真值中夹杂的奇异值(干扰量),保持真值序列的稳定性。

其主要思想是:当测量值累计达到 m 次(m 可根据具体需要设定大小)时则认为测量值正确,为测量真值。否则,还是以前次测量真值为准。

具体讲,取一个长度为 m 的向量:

$$l = \{ l(0), l(1) \dots, l(m-1) \}$$

变量 k 为最近一次的测量真值。进行 $m+1$ 次测量后得到测量值 q , 更新向量 l :

$$l(0) = l(1), l(1) = l(2), \dots,$$

$$l(m-2) = l(m-1), l(m-1) = q$$

比较向量 l 中的 m 个值是否相等,若:

$$l(0) = l(1) = \dots = l(m-1)$$

则本次测量值 q 即为测量真值 k_{m+1}

$$k_{m+1} = q$$

若不等,则测量真值为最近一次测量真值:

$$k_{m+1} = k_m$$

此方法的缺点是对信号变化的响应有所延迟,但延迟量很小,一般系统可忽略。

实现上述滤波方法的程序如下:

```

# define M 10 // 设定 m 的长度
int lxx[ M+ 1]; // 定义基准向量, lxx[0] 为存放最近
一次测量真值
void lxxlb(int q) // q 为测量值
{ int i= 0, k= 0;
for( i= 1; i< M; i+ + )
{ lxx[ i] = lxx[ i+ 1];
}
lxx[ M] = q;

```

```

for( i= M; i> 0; i- - )
{ if( x= lxx[ i] ) k+ + ;
}
if( k> = M)
{ lxx[ 0] = q;
return q; // 本次测量值 q 为测量真值, 返回
}
else
return lxx[ 0]; // 本次测量值 q 不可用, 取最近一次测量
真值, 返回
}

```

2 结束语

数字滤波的方法是很多,从某种角度上讲,只要能利用软件将采样值中的误差剔除,获得正确的采样值都可称为数字滤波。在程序编写中,我们还要考虑处理器的速度和存储空间限制,尽量编写节约 CPU 周期和存储空间的代码。如在均值滤波中,如要得到的采样值需为浮点型数据。不必每次采样后,将采样值 X_i 换算为浮点数,再求均值,可在求均值后再将 Y 换算为浮点型。这样,循环体内的运算为整形数据的累加运算,而不是浮点型数据的换算,可大大节约 CPU 周期,提高实时性。

上述介绍的几种数字滤波方法,都经过笔者的实际应用,效果是十分明显的。

参考文献

- 1 杨剑,姚志诚,刘光斌.一种简单的数字滤波方法.仪器仪表学报,2004,(3):增刊
- 2 刘光斌,姚志诚,刘东.单片机实用抗干扰技术.北京:人民邮电出版社,2003.
- 3 余锡存,曹国华.单片机原理及接口技术.西安:西安电子科技大学出版社,2000.

作者简介

杨剑,男,(1980年生),第二炮兵工程学院导航、制导与控制专业硕士研究生,研究方向为智能仪器及控制、系统可靠性与电磁环境控制。

刘光斌,男,(1963年生),第二炮兵工程学院教授,博士生导师,研究方向为智能仪器及控制、电磁兼容、系统可靠性与电磁环境控制。