电能质量分析中的 FFT 算法的改进与实现

李齐礼[1]　季振山[1]　舒双宝[2]

（1.中科院等离子体物理研究所，合肥 230031;2.合肥工业大学仪器科学与光电工程学院 230009）

摘要：电能质量分析中要求 *FFT* 算法具有更高的效率，而传统 *FFT* 算法在进行蝶形运算之前需要倒序排列，会产生较大的时间开销，本文提出了一种不需倒序的 FFT 算法，避免了地址倒序所带来的时间效率的开销。通过 C 语言实现了该算法，与 Matlab 自带的 FFT 算法进行了对比，仿真结果表明该算法是正确而高效的。

关键词：FFT; 信号处理; 算法;不倒序; 电能质量

中图分类号：TP301

Improvement and Implementation of FFT Algorithm for Power quality analysis

LI Qi-li[1] , Ji Zhenshan[1],Shu Shuangbao[2]

(1.Institute of Plasma Physics Chinese Academy of Sciences , Hefei , 230031; 2.School of Instrumentation and Opto-Electronics Engineering, Hefei University of Technology ,Hefei 230009,China)

Abstract: More efficient FFT algorithm is required in power quality analysis, while the traditional FFT algorithm have a greater time cost because of making butterfly in inverse-order. In order to boost real-time power quality analysis, this paper proposes a non-inversed order algorithm of FFT, which means much efficiency and space cost cuts. Compared with FFT algorithm supplied by matlab math library, the above algorithm realized in c language proved correct and highly efficient by simulation result.

Keywords: FFT, Signal Processing, Algorithm, Non Inversed Order, Power Quality

0 引言

电能质量分析很重要的方面是对引起电能质量问题的信号进行分析与处理[1]。傅里叶变换是信号分析处理十分有效的工具，但是进行电能质量分析[2-3]处理的一般是嵌入式设备，这些硬件要求 *FFT* 具有更高的效率。近几年来改进的 *FFT* 算法[4-6]大部分是基于蝶形运算进行改进，从减少蝶形运算的次数出发，降低了乘法的次数，提高了算法的效率。但是，这些算法有一个共同的特点，就是做蝶形运算之前需要对运算数据进行倒序处理，需要进行 $O(N)$ 量级[7]的数据交换，这必然带来较大时间效率上的开销。本文从数学的角度分析了 *FFT* 的特点，发现在增加一倍空间的情况下，可以不用对运算数据进行倒序处理，提出了一种不需地址倒序的改进 *FFT* 算法。

1　改进的 *FFT* 算法

离散傅里叶变换归结为计算：

$$X_j = \sum_{k=0}^{N-1} x_k w^{jk} \quad (j = 0,1,\cdots,N-1)$$
$$w = e^{-i\frac{2\pi}{N}} \ (i\text{为虚数单位}) \tag{1.1}$$

性质 1:如果 $m \equiv r(\bmod N), 0 \le r < N$ ，则有[8]

$$w^m = w^{Nq+r} = (w^N)^q w^r = w^r \text{。} \tag{1.2}$$

因此计算 w^m 的时候 m 可以用同余数 r 代替，实际上，在所有 $w^{jk}(k, j = 0,1,\cdots,N-1)$ 中，只有 N 个不同的值 ，特别当 $N = 2^p$ 时，只有 $N/2$ 个不同的值。因此可把同一个 w^r 对应的 x_k 相加后再相乘，这样就能大量减少乘法次数，下面给出具体推导过程($N = 2^p$):

为了推导的方便，将 k, j 用二进制表示为:

$$k = \sum_{m=0}^{p-1} k_m 2^m = (k_{p-1},\cdots,k_1,k_0)$$
$$j = \sum_{m=0}^{p-1} j_m 2^m = (j_{p-1},\cdots,j_1,j_0) \tag{1.3}$$

项目基金:科技部 973 项目(NO.2009GB103000),先进等离子体控制方法与试验研究

根据性质 1,把式(1.3)代入计算 $w^{j\bullet k}$ 有

$$\begin{aligned} w^{j\bullet k} &= w^{\sum_{n=0}^{p-1}\sum_{m=0}^{p-1} j_n 2^n k_m 2^m} \\ &= w^{\sum_{n=0}^{p-1} j_n \sum_{m=0}^{p-1} k_m 2^{m+n}} \\ &= w^{\sum_{n=0}^{p-1} j_n (\sum_{m=0}^{p-1} k_m 2^{m+n})} \\ &= \prod_{n=0}^{p-1} w^{j_n \bullet \sum_{m=0}^{p-1} k_m 2^{m+n}} \\ &= \prod_{n=0}^{p-1} w^{j_n (k_{p-n-1},\cdots,k_0,\underbrace{0,\cdots,0}_{n个0})} \end{aligned}$$

把 $w^{j\bullet k}$ 代入式(1.1)中，j 用式(1.3)的二进制表示，因此 X_j 即 $X(j_{p-1},\cdots,j_1,j_0)$ 可用下式来计算:

$$\begin{aligned} &X(j_{p-1},\cdots;j_1,j_0) \\ &=\sum_{k_0=0}^{1}\cdots\sum_{k_{p-1}=0}^{1} x\left(k_{p-1},\cdots,k_1,k_0\right)\prod_{n=0}^{p-1} w^{j_n(k_{p-n-1},\cdots;k_0,\underbrace{0,\cdots;0}_{n个0})} \\ &=\sum_{k_0=0}^{1}\cdots\sum_{k_{p-2}=0}^{1}[\sum_{k_{p-1}=0}^{1} x\left(k_{p-1},\cdots;k_1,k_0\right)w^{j_0(k_{p-1},\cdots;k_0)}]w^{j_1(k_{p-2},\cdots;k_0,0)} \end{aligned} \tag{1.4}$$

引入下面的记号:

$$A_0\left(k_{p-1},\cdots,k_2,k_1,k_0\right)=x\left(k_{p-1},\cdots,k_1,k_0\right)$$

$$A_1\left(k_{p-2},\cdots,k_1,k_0,j_0\right)=\sum_{k_{p-1}=0}^{1} A_0\left(k_{p-1},\cdots,k_2,k_1,k_0\right)w^{j_0(k_{p-1},\cdots,k_0)}$$

$$A_2\left(k_{p-3},\cdots,k_0,j_1,j_0\right)=\sum_{k_{p-2}=0}^{1} A_1\left(k_{p-2},\cdots,k_1,k_0,j_0\right)w^{j_1(k_{p-2},\cdots,k_0,0)}$$

……

$$A_p\left(j_{p-1},\cdots,j_1,j_0\right)=\sum_{k_0=0}^{1} A_{p-1}\left(k_0,j_{p-2}\cdots,j_1,j_0\right)w^{j_{p-1}(k_0,0,\cdots,0)}$$

(1.5)

则式(1.4)可表示为

$$X(j_{p-1},\cdots,j_1,j_0)=A_p\left(j_{p-1},\cdots,j_1,j_0\right)$$

式(1.5)说明计算所有 X_j 只需要分为 p 步，每步需要 $2N$ 次复数乘法，总共需要 $2pN$ 次复数乘法。

注意到[8]

$$w^{j2^{p-1}}=w^{jN/2}=(-1)^j$$

对式(1.5)的通式进一步化简(其中 $q=1,...,p$)

$$\begin{aligned} &A_q\left(k_{p-q-1},\cdots,k_0,j_{q-1}\cdots j_0\right) \\ &=\sum_{k_{p-q}=0}^{1} A_{q-1}\left(k_{p-q},\cdots,k_0,j_{q-2}\cdots j_0\right)w^{j_{q-1}(k_{p-q},\cdots,k_0,\underbrace{0\cdots0}_{q-1个0})} \\ &=\sum_{k_{p-q}=0}^{1} A_{q-1}\left(k_{p-q},\cdots,k_0,j_{q-2}\cdots j_0\right)\left(w^{j_{q-1}\left[k_{p-q}\bullet 2^{p-1}+(0,k_{p-q-1},\cdots,k_0,\underbrace{0\cdots0}_{q-1个0})\right]}\right) \\ &=\sum_{k_{p-q}=0}^{1} A_{q-1}\left(k_{p-q},\cdots,k_0,j_{q-2}\cdots j_0\right)\left(w^{j_{q-1}\bullet k_{p-q}\bullet 2^{p-1}}w^{j_{q-1}(0,k_{p-q-1},\cdots,k_0,\underbrace{0\cdots0}_{q-1个0})}\right) \\ &=\sum_{k_{p-q}=0}^{1} A_{q-1}\left(k_{p-q},\cdots,k_0,j_{q-2}\cdots j_0\right)\left((-1)^{j_{q-1}\bullet k_{p-q}}w^{j_{q-1}(0,k_{p-q-1},\cdots,k_0,\underbrace{0\cdots0}_{q-1个0})}\right) \\ &=\begin{bmatrix} A_{q-1}\left(0,k_{p-q-1},\cdots,k_0,j_{q-2}\cdots j_0\right)+ \\ A_{q-1}\left(1,k_{p-q-1},\cdots,k_0,j_{q-2}\cdots j_0\right)(-1)^{j_{q-1}} \end{bmatrix}w^{j_{q-1}(0,k_{p-q-1},\cdots,k_0,\underbrace{0\cdots0}_{q-1个0})} \end{aligned}$$

又因为二进制位 j_{q-1} 取值为 0 和 1 两种情况，所以上式可表示如下所示:

$$\begin{aligned} &A_q\left(k_{p-q-1},\cdots;k_0,0,j_{q-2}\cdots j_0\right) \\ &=A_{q-1}\left(0,k_{p-q-1},\cdots;k_0,j_{q-2}\cdots j_0\right)+A_{q-1}\left(1,k_{p-q-1},\cdots;k_0,j_{q-2}\cdots j_0\right) \\ &A_q\left(k_{p-q-1},\cdots;k_0,1,j_{q-2}\cdots j_0\right) \\ &=\left[A_{q-1}\left(0,k_{p-q-1},\cdots;k_0,j_{q-2}\cdots j_0\right)-A_{q-1}\left(1,k_{p-q-1},\cdots;k_0,j_{q-2}\cdots j_0\right)\right]w^{(0,k_{p-q-1},\cdots;k_0,\underbrace{0\cdots0}_{q-1个0})} \end{aligned} \tag{1.6}$$

把式(1.6)中的二进制表示还原为十进制表示，令

$$k=(k_{p-q-1},\cdots,k_0)=k_{p-q-1}2^{p-q-1}+\cdots+k_0 即 k=0,1,\cdots,2^{p-q}-1$$

$$j=\left(j_{q-2}\cdots j_0\right)=j_{q-2}2^{q-2}+\cdots+j_0 即 j=0,1,\cdots,2^{q-1}-1$$

得到改进的 FFT 的计算公式

$$\begin{aligned} &A_q\left(k2^q+j\right)=A_{q-1}\left(k2^{q-1}+j\right)+A_{q-1}\left(k2^{q-1}+j+2^{p-1}\right) \\ &A_q\left(k2^q+j+2^{q-1}\right)=\left[A_{q-1}\left(k2^{q-1}+j\right)-A_{q-1}\left(k2^{q-1}+j+2^{p-1}\right)\right]w^{k2^{q-1}} \end{aligned} \tag{1.7}$$

其中

$$q=1,\cdots,p$$
$$k=0,1,\cdots,2^{p-q}-1$$
$$j=0,1,\cdots,2^{q-1}-1$$

该FFT算法的特点：

(1)不需要逆序排列处理，节省了N次数据交换的时间，大大提高了效率。

(2)计算只有两重循环，外循环q由1到p，内循环k由0计算到$2^{p-q}-1$，j由0计算到$2^{q-1}-1$，简化了程序设计。

(3)计算量：加法次数：pN，乘法次数：$pN/2$

(4)所需空间增加一倍。

2 算法流程

(i)定义数组$A_1(N),A_2(N),w(N/2)$;

(ii)将已知的记录复数数组$\{x_k\}$输入到单元$A_1(k)(k=0,1,\cdots N-1)$中;

(iii)计算$w^m=e^{-i\frac{2\pi}{N}m}=\cos\left(\frac{2\pi}{N}m\right)-i\bullet\sin\left(\frac{2\pi}{N}m\right)$，存放在单元$w(m)(m=0,1,\cdots N/2-1)$中;

(iv)q循环$(1,\cdots,p)$，若q为奇数执行步骤(v),否则执行步骤(vi);

(v) k循环$\left(0,\cdots,2^{p-q}-1\right)$，$j$循环$\left(0,\cdots,2^{q-1}-1\right)$，计算:

$$A_2\left(k2^q+j\right)=A_1\left(k2^{q-1}+j\right)+A_1\left(k2^{q-1}+j+2^{p-1}\right)$$
$$A_2\left(k2^q+j+2^{q-1}\right)=\left[A_1\left(k2^{q-1}+j\right)+A_1\left(k2^{q-1}+j+2^{p-1}\right)\right]w^{k2^{q-1}}$$

循环结束，转到步骤(vii);

(vi) k循环$\left(0,\cdots,2^{p-q}-1\right)$，$j$循环$\left(0,\cdots,2^{q-1}-1\right)$，计算:

$$A_1\left(k2^q+j\right)=A_2\left(k2^{q-1}+j\right)+A_2\left(k2^{q-1}+j+2^{p-1}\right)$$
$$A_1\left(k2^q+j+2^{q-1}\right)=\left[A_2\left(k2^{q-1}+j\right)+A_2\left(k2^{q-1}+j+2^{p-1}\right)\right]w^{k2^{q-1}}$$

循环结束，转到步骤(vii);

(vii)若$q=p$转到步骤(viii)，否则$q+1\to q$转到步骤(iv);

(viii)q循环结束，若p为偶数，将$A_1(j)\to A_2(j)$，则$X_j=A_2(j)(j=0,1,\cdots,N-1)$即为所求。

3 仿真实验

(1)利用$VS2010$实现了该改进的FFT算法，把结果写入二进制文件$fft.dat$中，然后在$Matlab$中读$fft.dat$，画出幅值-频率图像，并且与$Matlab$自带的FFT函数产生的结果进行了对比，采样率设置为$12800SPS$，原始信号$x=\sin(2\pi\times50\times t)+2\sin(2\pi\times800\times t)+\sin(2\pi\times1200\times t)+3\sin(2\pi\times4000\times t)$算法中分别进行了256、512、1024点$FFT$的比较。仿真结果分别如图3.1，3.2，3.3所示，该算法与$Matlab$自带的FFT算法结果差值很小(256点FFT相差10^{-4}数量级以内，512点FFT相差10^{-3}数量级以内，1024点FFT相差10^{-2}数量级以内)。从仿真结果可知，该改进的FFT算法是正确的。

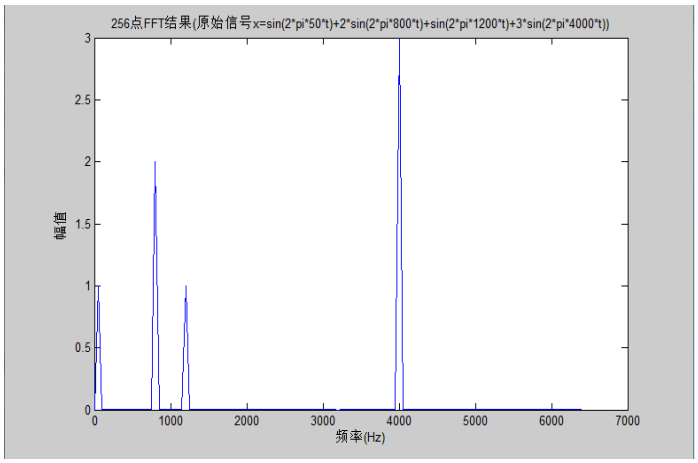
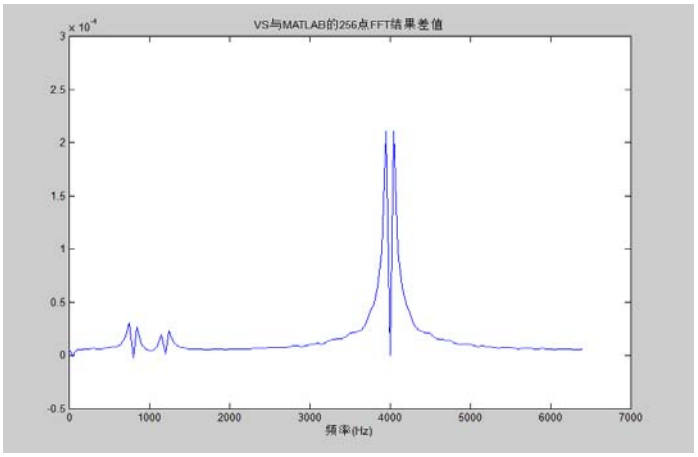


图 3.1 256 点 FFT 结果



VS 与 Matlab 差值

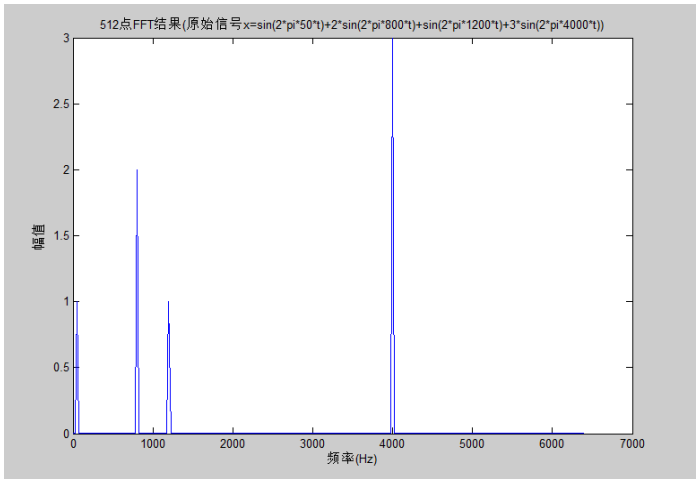
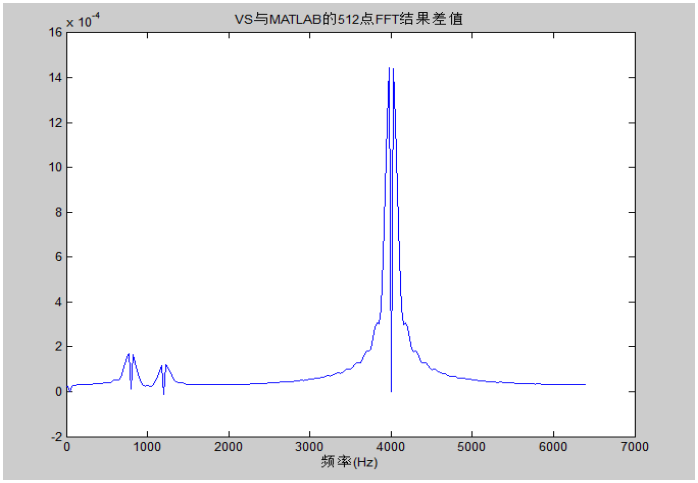


图 3.2 512 点 FFT 结果



VS 与 Matlab 差值

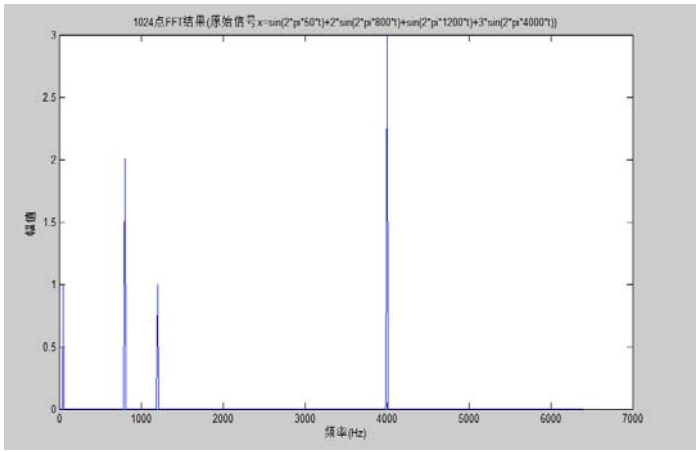
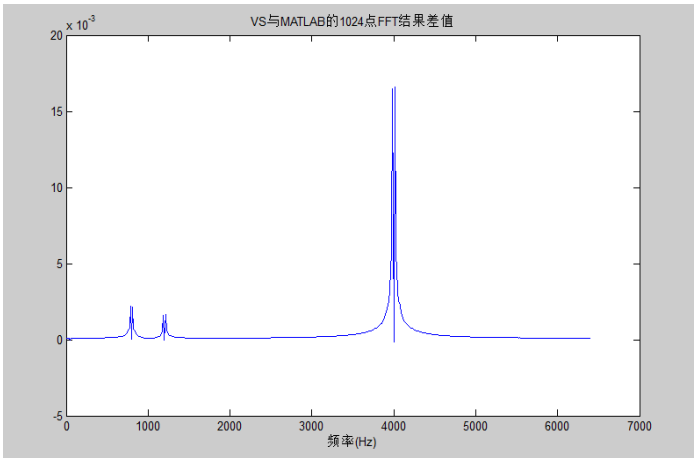


图 3.3 1024 点 FFT 结果



VS 与 Matlab 差值

(2)时间效率对比。在 PC 机(硬件配置：Core i3　3.06GHz，内存 4GB)Win32 环境下运行改进 FFT 和普通 FFT[9][10]的 C 语言程序，利用 C 库函数 clock()统计了两种算法运行 1000 次的时间，如表 3.1 和图 3.4 所示.从图表的结果可以看到，改进 FFT 算法大大改善了时间效率。

表 3.1 改进 FFT 与普通 FFT 算法运行 1000 次的时间(单位毫秒)

点数	64	128	256	512	1024	2048	4096
普通 FFT（ms）	291	856	1904	4261	9296	20229	43437
改进 FFT（ms）	27	37	84	148	312	634	1374

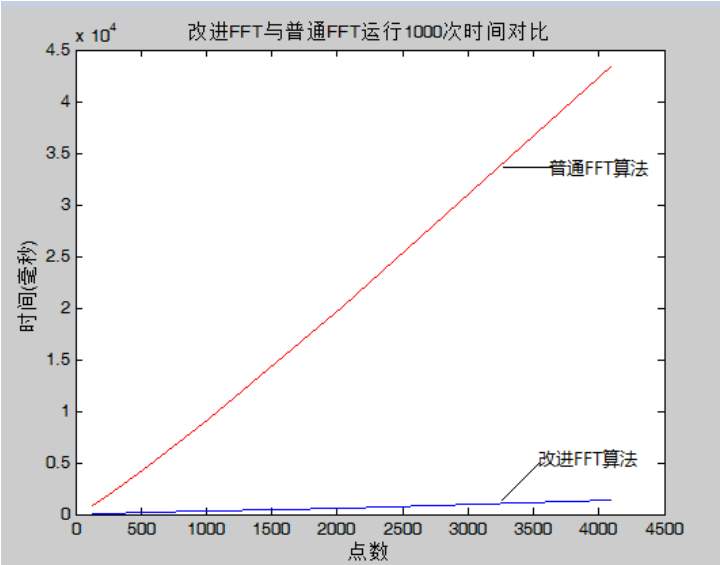


图 3.4 改进 FFT 与普通 FFT 算法运行 1000 次所需时间对比

4 总结

本文提出的不需地址倒序的 *FFT* 算法，是对普通蝶形运算的改进，在一定程度上改进了算法的效率，并且大大降低了算法实现的复杂性。本文对该算法给出了严格的数学推导，利用 C 语言实现了该算法，利用 Matlab 进行了算法验证，并且对两种算法的时间性能进行了对比。该算法十分适合电能质量分析中的信号处理，能大大提高处理实效性。

参考文献：

[1] 肖湘宁.电能质量分析与控制[M].北京:中国电力出版社，2004:10-24.

Xiao Xiang-ning. Analysis and Control of Power quality [M]. Beijing: China Electric Power Press, 2004:10-24.

[2] 舒双宝，罗家融，王勤湧等.基于 DSP 和 ARM 便携式电能质量监测系统的设计与实现[J]. 电力系统保护与控制, 2010,38(24):185-189.

Shu Shuang-bao, Luo Jia-rong, Wang Qin-yong et al. Design and implementation of a portable power quality monitoring device based on DSP and ARM[J].Power System Protection and Control, 2010,38(24):185-189.

[3] 陈国磊，舒双宝，季振山.电能质量监测高速数据采集系统的设计和实现[J]. 电力系统保护与控制，2009,37(3):70-72.

Chen Guo-lei, Shu Shuang-bao, Ji Zhen-shan. Design and implementation of data acquisition system for power quality monitoring[J].Power System Protection and Control, 2009,37(3):70-72.

[4] Santoso S. Power Quality Assessment via Wavelet Transform Analysis[J].IEEE Trans on Power Delivery, 1996, 11(2): 924 - 930 .

[5] ZHANG Fu-sheng, GENG Zheng-xing, YUAN Wei. The algorithm of interpolating windowed FFT for harmonic analysis of electric power system[J] .IEEE Trans on Power Delivery, 2001,162, 16(2) :160-164 .

[6] 王超,方勇,张倩.基于修正离散傅里叶变换的频域卷积混合盲分离[J].数据采集与处理，2009，24(5):556-562.

Wang Chao, Fang Yong, Zhang Qian .Frequency Domain Convolutive Blind Separation Based on Modified Discrete Fourier Transform[J]. Journal of Data Acquisition and Processing, 2009,24(5):556-562.

[7] 王兆华,黄翔东. 基于全相位谱分析的相位测量原理及其应用[J] 数据采集与处理，2009,24(6) :777-782.

Wang Zhao-hua, Huang Xiang-dong. Principle of Phase Measurement and Its Application Based on All-Phase Spectral Analysis[J]. 2009,24(6）:777-782.

[8] 李庆扬,王能超,周大义.数值分析[M].武汉:华中科技大学出版社，2006:55-58.

Li Qing-yang, Wang Neng-chao, Zhou Da-yi Numerical Analysis [M]. Wuhan: Huazhong University of Science and Technology Press, 2006:55-58.

[9] THOMAS G. Interpolation algorithms for discrete fourier transform of weighed signals[J].IEEE Trans IM, 1983,322, 32(2) :350-355.

[10] 刘鲭洁,陈桂明,刘晓方等.FFT 和小波变换在信号降噪中的应用[J].数据采集与处理，2009,24（增刊）：58-60.

Liu Qing-jie, Chen Gui-ming, Liu Xiao-fang et al. Application of FFT and Wavelet in Signal Denoising[J]. Journal of Data Acquisition and Processing, 2009, 24(suppl）:58-60.

作者简介:

李齐礼(1985-)，男，硕士研究生，研究方向：嵌入式数据采集与信号处理；qlli@ipp.ac.cn

季振山(1963-)，男，研究员，硕士生导师，研究方向：计算机控制和数据采集系统的研究。

舒双宝(1981-)，男，讲师，博士，研究方向：计算机采集与控制、模式识别。