

# 基于 CIC 滤波器原理的音频信号快速重采样算法

张江安<sup>1</sup> 杨洪柏<sup>2</sup>

<sup>1</sup>(上海工程技术大学 上海 200437)

<sup>2</sup>(上海开放大学理工学院 上海 200433)

**摘要** 针对现有音频信号重采样算法普遍具有的计算开销大的缺点,提出一种基于级联积分器与梳状(CIC)滤波器原理的音频信号快速重采样算法,应用于计算资源有限的嵌入式系统中。构造同时包含内插和抽取环节的三级 CIC 重采样滤波器,并根据其工作过程提出一种基本的音频信号重采样算法。推导由积分器寄存器值计算重采样值的公式,替代基本重采样算法中耗时的内插循环操作,从而得到音频信号快速重采样算法。该快速算法不需要存储滤波器系数,计算开销极小,在满足奈奎斯特采样定理的前提下可实现任意采样率转换,极大地提高了嵌入式音频重采样系统的性价比。

**关键词** 嵌入式系统 CIC 滤波器 音频信号 重采样 快速算法 性价比

**中图分类号** TN912.3 TP319 **文献标识码** A **DOI**:10.3969/j.issn.1000-386x.2019.07.047

## A FAST RESAMPLING ALGORITHM OF AUDIO SIGNAL BASED ON THE PRINCIPLE OF CIC FILTER

Zhang Jiang'an<sup>1</sup> Yang Hongbai<sup>2</sup>

<sup>1</sup>(Shanghai University of Engineering Science, Shanghai 200437, China)

<sup>2</sup>(School of Science and Technology, Shanghai Open University, Shanghai 200433, China)

**Abstract** Aiming at the shortcoming that the existing audio signal resampling algorithm has great computational overhead, a fast resampling algorithm of audio signal based on the principle of Cascade Integrator and Comb(CIC) filter is proposed. We applied it to the embedded system with limited computing resources. We constructed a three-stage CIC resampling filter containing both interpolation and decimation links, and proposed a basic audio signal-resampling algorithm according to its working process. Then, the formula for calculating the resampling value from the integrator register value was deduced, which replaced the time-consuming interpolation loop operation in the basic resampling algorithm, and the fast resampling algorithm of the audio signal was obtained. The fast algorithm does not need the storage filter coefficient, and has very small calculation overhead. It can realize arbitrary sampling rate conversion under the premise of satisfying the Nyquist sampling theorem, and greatly improves the performance price ratio of the embedded audio resampling system.

**Keywords** Embedded system CIC filter Audio signal Resampling Fast algorithm Performance price ratio

## 0 引言

在日常生活和工作中嵌入式音频播放系统被广泛使用,例如便携式音箱、MP3 播放器等。这些嵌入式系

统往往具有有限的硬件资源,例如作为主控芯片的数字信号处理器(DSP)、微控制器(MCU)等具有有限的存储空间和计算能力,而音频输出端口往往仅支持少数信号采样频率。因此,在这些音频播放系统中,常常需要采用信号重采样算法<sup>[1]</sup>,以软件方式实时改变输

入音频信号采样率,以解决音频解码器(decoder)与音频信号采样率之间不匹配的问题。其中,采样频率转换倍率一般为分数 $\frac{M}{N}$ ( $M$ 、 $N$ 为互质正整数)。常用的分数倍信号重采样方法包括有限冲激响应(FIR)滤波器法<sup>[2]</sup>、多相滤波器组法<sup>[3]</sup>、Farrow 结构滤波器<sup>[4-7]</sup>法等。其中多相滤波器组法处理过程复杂,不利于软件实现。FIR 滤波器法和 Farrow 结构滤波器法均需要较多数据存储空间用以存储滤波器系数。当  $N$  和  $M$  变化时,滤波器系数也发生改变,存储空间需要相应地增加。另一方面,这些重采样方法需要大量的加法和乘法运算,计算开销很大。因此,上述方法不适合应用于计算资源有限的嵌入式音频系统中。

面对上述问题,寻求一种能够应用于嵌入式音频系统的简便而高效的信号重采样算法显得尤为必要。Hogenauer 提出的级联积分器和梳状(Cascaded Integrator-Comb, CIC)滤波器<sup>[8-11]</sup>不仅具有良好的低通特性,而且由于其结构简单、无乘法运算、不需存储滤波器系数等特点,在多采样率信号处理中被广泛应用。在已有的应用中,CIC 滤波器较多地被用于信号抽取或内插处理,以整数倍降低或升高信号采样率。文献[12]在通信系统中以硬件电路形式将 CIC 滤波器用于特定分数倍信号采样率变换,但它并未给出重采样算法,无法在嵌入式系统中应用。本文面向嵌入式系统中音频信号分数倍采样率变换的实际需求,根据 CIC 滤波器工作原理,提出了一种快速的音频信号重采样算法。首先在分析 CIC 滤波器结构特点的基础上,构造包含内插和抽取操作的三级 CIC 重采样滤波器,并根据其工作过程中的信号流得到基本的重采样算法。然后通过简化内插和抽取操作,减小运算开销,得到高效、快速的音频信号重采样算法。该算法不仅能够保持较好的音频质量,而且内存需求少,计算速度快。

## 1 三级 CIC 重采样滤波器的构造

CIC 滤波器由级联梳状滤波器、级联积分器、内插或抽取环节、增益环节构成。图 1 为三级级联梳状滤波器,其中  $C_0$ 、 $C_1$ 、 $C_2$  分别为各梳状滤波器寄存器的值, $S_1$ 、 $S_2$ 、 $S_3$  分别为各梳状滤波器的输出值。图 2 为三级级联积分器,其中  $D_1$ 、 $D_2$ 、 $D_3$  分别为各积分器寄存器的值, $V_1$ 、 $V_2$ 、 $V_3$  分别为各积分器的输出值。图 3 为三级 CIC 内插滤波器,它可实现将信号采样率提升  $N$  倍。其中  $N$  倍内插环节所执行的操作为插零操作;

其增益环节值为 $\frac{1}{N^2}$ 。图 4 为三级 CIC 抽取滤波器,它可实现信号  $M$  倍抽取,即采样率降为原采样率的 $\frac{1}{M}$ 。

其增益环节值为 $\frac{1}{M^3}$ 。

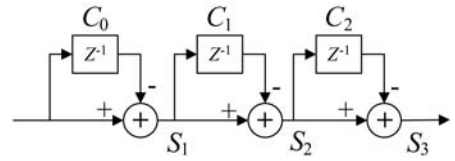


图 1 三级级联梳状滤波器

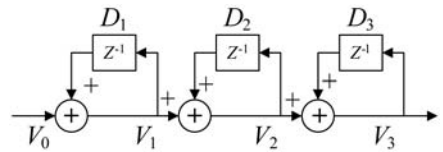


图 2 三级级联积分器

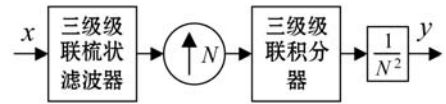


图 3 三级 CIC 内插滤波器

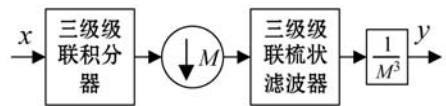


图 4 三级 CIC 抽取滤波器

对于变化倍数为 $\frac{N}{M}$ 的信号重采样操作,可通过先内插  $N$  倍,再进行  $M$  倍抽取的方式实现。假设信号重采样操作满足奈奎斯特采样定理,显然,这样的操作可以通过图 5 所示的三级 CIC 重采样滤波器结构实现。



图 5 三级 CIC 重采样滤波器

## 2 音频信号基本重采样算法

显然,CIC 滤波器级联级数越多,低通滤波效果越好,但滤波器结构也越复杂。三级 CIC 重采样滤波器结构较为简单,且具有较强低通性能,可满足音频信号重采样处理的要求。根据图 1、图 2、图 5,采用算法形式描述三级 CIC 重采样滤波器的工作过程,得到如下音频信号基本重采样算法:

```

Input:   x //输入信号序列
Output:  y //输出信号序列
1.  t←0, k←0 //控制变量初始化
2.  D1←0, D2←0, D3←0 //寄存器值初始化
    C1←0, C2←0, C3←0 //寄存器值初始化

```

```

3. For  $i$  from 1 to  $L$  do //逐点输入信号
3.1  $S_1 \leftarrow x(i) - C_0$  //梳状滤波器计算
3.2  $S_2 \leftarrow S_1 - C_1$  //梳状滤波器计算
3.3  $S_3 \leftarrow S_2 - C_2$  //梳状滤波器计算
3.4  $C_0 \leftarrow x(i), C_1 \leftarrow S_1, C_2 \leftarrow S_2$  //寄存器值更新
3.5 For  $j$  from 1 to  $N$  do //内插循环
3.5.1  $t \leftarrow t + 1$ 
3.5.2 If  $j = 1$  then:  $V_0 \leftarrow S_3$  //数值传递
    Else:  $V_0 \leftarrow 0$  //零值内插
3.5.3  $V_1 \leftarrow V_0 + D_1$  //积分器计算
3.5.4  $V_2 \leftarrow V_1 + D_2$  //积分器计算
3.5.5  $V_3 \leftarrow V_2 + D_3$  //积分器计算
3.5.6 If  $t = M$  then: //抽取操作
     $k \leftarrow k + 1, y(k) \leftarrow V_3 / N^2, t \leftarrow 0$ 
3.5.7  $D_1 \leftarrow V_1, D_2 \leftarrow V_2, D_3 \leftarrow V_3$  //寄存器值更新
4. Return( $y$ )
    
```

在上述基本算法中,  $L$  为输入音频信号采样数。  $t$  是一个决定信号抽取时刻的时钟计数器, 其值在  $0 \sim M$  之间循环变化。在算法执行之初将其初始化为 0。随后每当内插环节插入一个样值 0,  $t$  值增 1。算法运行中, 每当  $t = M$  条件成立时, 抽取  $V_3$  值并经增益处理后作为算法输出值, 同时  $t$  值清零。显然, 由于内插循环的存在, 上述基本重采样算法需要耗费大量的运算时间, 效率较低。

### 3 音频信号快速重采样算法

对上述基本重采样算法的内插循环进行分析。当内插循环计数器  $j = 1$  时, 级联梳状滤波器的输出值  $S_3$  被传至级联积分器,  $V_0 = S_3, V_1 = V_0 + D_1, V_2 = V_1 + D_2, V_3 = V_2 + D_3$ 。将此时  $V_1, V_2, V_3$  的值分别记为  $V'_1, V'_2, V'_3$ 。根据步骤 3.5.7, 显然,  $D_1 = V'_1, D_2 = V'_2, D_3 = V'_3$ 。

随后内插循环进入插零操作期, 即  $V_0$  始终被赋 0 值。因此, 当  $j = 2$  时,  $V_1 = D_1 = V'_1, V_2 = V_1 + D_2 = V'_1 + V'_2, V_3 = V_2 + D_3 = V'_1 + V'_2 + V'_3$ 。

根据步骤 3.5.7,  $D_1, D_2, D_3$  的值随后也被更新:  $D_1 = V'_1, D_2 = V'_1 + V'_2, D_3 = V'_1 + V'_2 + V'_3$ 。

不断重复上述过程可知, 对于任意  $j \leq N$ , 可得:  $V_1 = V'_1, V_2 = (j-1)V'_1 + V'_2, V_3 = [(j-1) + \dots + 2 + 1]V'_1 + (j-1)V'_2 + V'_3 = \frac{1}{2}j(j-1)V'_1 + (j-1)V'_2 + V'_3$

根据步骤 3.5.6, 当抽取操作发生时:

$$y(k) = \frac{V_3}{N^2} = \frac{j(j-1)V'_1 + 2(j-1)V'_2 + 2V'_3}{2N^2} \quad (1)$$

由此可见, 重采样输出值  $y(k)$  可由当前内插循环  $V'_1, V'_2, V'_3$  的值以及循环标号  $j$  计算得到, 从而使内插

循环得以替代。

因此, 可取消内插循环操作, 改为通过式(1)直接计算重采样输出值。为适应此变化,  $t$  值更新方式需作相应改变。每当系统输入一个采样值  $x(i)$  时,  $t$  增加  $N$ 。每当系统输出一个重采样  $y(k)$  时,  $t$  减小  $M$  (参见步骤 3.5.6)。

当  $t \geq M$  时, 将执行重采样信号抽取操作。根据图 6 可知, 此时内插循环计数器为:

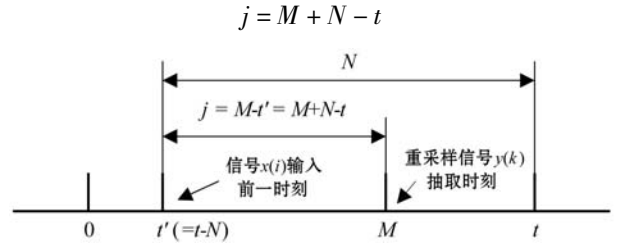


图 6 重采样时刻内插循环计数器的计算  
参照式(1), 得到如下重采样计算公式:

$$y(k) = \frac{j(j-1)V_1 + 2(j-1)V_2 + 2V_3}{2N^2} \quad (2)$$

式中:  $j = M + N - t$ 。由于内插循环被取消,  $V'_1, V'_2, V'_3$  可分别由  $V_1, V_2, V_3$  替代。每当完成一次重采样计算, 令:

$$t \leftarrow t - M \quad (3)$$

随后检查  $t \geq M$  是否仍然成立。若成立, 则说明仍然有信号抽取操作等待执行。重复式(2)、式(3)的计算和操作, 直至  $t < M$  时为止。

综上, 提出基于三级 CIC 重采样滤波器的音频信号快速重采样算法如下:

```

Input:    $x$  //输入信号序列
Output:   $y$  //输出信号序列
1.  $t \leftarrow 0, k \leftarrow 0$  //控制变量初始化
2.  $D_1 \leftarrow 0, D_2 \leftarrow 0, D_3 \leftarrow 0$  //寄存器值初始化
    $C_1 \leftarrow 0, C_2 \leftarrow 0, C_3 \leftarrow 0$  //寄存器值初始化
3. For  $i$  from 1 to  $L$  do //逐点输入信号
3.1  $S_1 \leftarrow x(i) - C_0$  //梳状滤波器计算
3.2  $S_2 \leftarrow S_1 - C_1$  //梳状滤波器计算
3.3  $S_3 \leftarrow S_2 - C_2$  //梳状滤波器计算
3.4  $C_0 \leftarrow x(i), C_1 \leftarrow S_1, C_2 \leftarrow S_2$  //寄存器值更新
3.5  $V_1 \leftarrow D_1 + S_3$  //积分器计算
3.6  $V_2 \leftarrow D_2 + V_1$  //积分器计算
3.7  $V_3 \leftarrow D_3 + V_2$  //积分器计算
3.8  $t \leftarrow t + N$  //控制变量更新
3.9 While  $t \geq M$  do //输出值计算循环
3.9.1  $k \leftarrow k + 1$ 
3.9.2  $j \leftarrow M + N - t$ 
3.9.3  $y(k) \leftarrow (j(j-1)V_1 + 2(j-1)V_2 + 2V_3) / (2N^2)$ 
3.9.4  $t \leftarrow t - M$  //t 值更新
    
```

3.10  $D_1 \leftarrow V_1$  //寄存器值更新(下同)  
 3.11  $D_2 \leftarrow (N-1)V_1 + V_2$   
 3.12  $D_3 \leftarrow N(N-1)V_1/2 + (N-1)V_2 + V_3$

4. Return( $y$ )

在上述快速算法中, While 循环为计算重采样输出值的循环。由于算法中  $N$  为常数, 在上述算法中, 可以将除以“ $2N^2$ ”运算转变为乘以“ $1/(2N^2)$ ”的运算, 以达到在嵌入式系统中简化计算的目的。从基本算法到快速算法,  $t$  值变化的步长由 1 变为  $N$ , 重采样计算速度得以加快。上述快速重采样算法的流程图如图 7 所示。

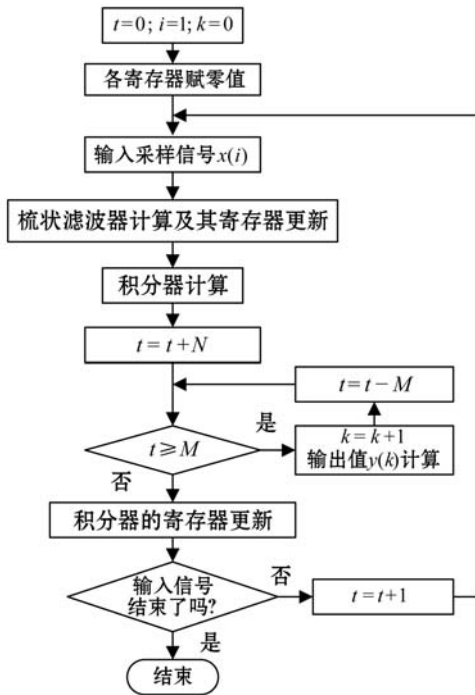


图7 快速重采样算法流程图

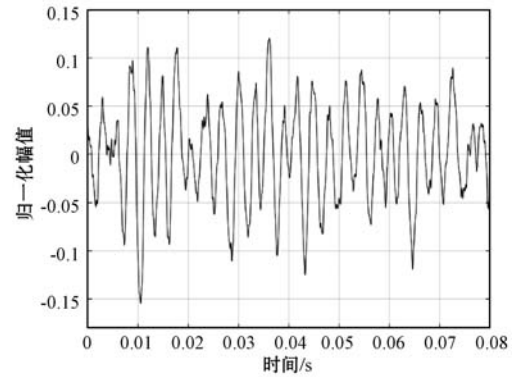
## 4 测试与分析

### 4.1 主观测试

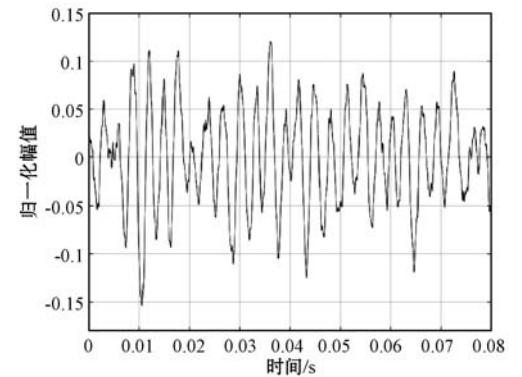
为验证上述音频信号快速重采样算法的有效性, 本文在 MATLAB 平台, 对音频信号快速重采样算法进行了大量测试。测试信号由多种类型的音频信号组成, 其中语音、歌曲、纯音乐各 40 段, 每段长约 3 分钟。信号原始频率包括 22.05 kHz、32 kHz、44.1 kHz、48 kHz 四种规格, 每种规格信号总数的 1/4。测试在四种不同的采样频率之间进行。通过试听可知, 采样频率转换后的音频可以较好地保持原有音频的质量。

图 8 显示了一段原始音频信号与相应的重采样音频信号波形。原始音频信号的采样率为 44.1 kHz, 经重采样处理后采样率变为 48 kHz。由图 8 可知, 音频

信号快速重采样算法能够较好地保持信号的细节, 从而保证了转换后的音频质量。



(a) 原始信号片段(采样率:44.1 kHz)



(b) 重采样信号(采样率:48 kHz)

图8 原始音频信号与重采样信号

### 4.2 误差分析

本文所提出的音频信号快速重采样算法构建在三级 CIC 滤波器基础上, 其通带增益特性可由下式计算<sup>[8]</sup>:

$$A(f) = \frac{1}{N^3} \left| \frac{\sin \pi f}{\sin \pi f / N} \right|^3 \quad (4)$$

式中:  $f$  为信号相对 CIC 内高采样频率的归一化频率。由式(4)可知,  $A(f)$  具有滚降特性。当  $f$  趋近 0 时,  $A(f)$  趋近 1, 信号的衰减很小。当  $f$  增大时, 信号的衰减增大。相对应地, 在基于 FIR 滤波器的重采样方法中, 由于 FIR 滤波器的通带通常被设计得非常平坦, 所得到重采样信号的失真将非常小。因此, 可将基于 FIR 滤波器重采样算法(以下简称为 FIR 算法)的输出音频作为频率转换后的基准音频, 对本文提出的基于 CIC 原理快速重采样算法(以下简称 CIC 算法)的精度进行对比分析。

FIR 算法由 MATLAB(R2016b 版本)软件中的重采样函数 resample 实现, 其中低通滤波器为 FIR 滤波器(通带纹波小于 0.1 dB)。其调用形式为  $[y, b] = \text{resample}(x, N, M)$ , 其中  $x, y$  分别为输入、输出信号,  $b$  为滤波器系数向量。设 CIC 算法的输出为  $\hat{y}$ , 为便于

分析,定义  $\hat{y}$  的信噪比为:

$$SNR = 10 \lg \frac{\sum_{n=1}^{L'} y^2(n)}{\sum_{n=1}^{L'} [y(n) - \hat{y}(n)]^2} \quad (5)$$

式中: $L'$ 为信号长度。

对 4.1 中实验数据进行误差分析,得到各种类别音频重采样信号的平均信噪比(如表 1 所示)。由表 1 可见,CIC 算法对包含语音、歌曲和纯音乐在内的各种音频信号均有较好的转换质量。

表 1 快速算法输出音频的信噪比

音频类别	平均信噪比/dB
语音	35.9
歌曲	36.3
纯音乐	37.5

### 4.3 计算开销分析

为了对算法的计算开销进行对比,在 ADI 公司 DSP 开发平台 VisualDSP ++ 3.5 上分别实现 CIC 算法、FIR 算法、Farrow 结构滤波器算法<sup>[4]</sup>(以下简称 Farrow 算法)。结果表明,CIC 算法的计算开销远远小于其他算法。以转换前后采样率分别为 44.1 kHz、48 kHz( $N = 160, M = 147$ )为例。表 2、表 3 分别列出了这三种算法的计算量和存储空间需求。表 3 中,ROM 用于存储滤波器系数,RAM 用于存储中间数据。当需要支持更多转换倍率时,CIC 算法不需要增加存储空间,而其他两种算法则需要增加 ROM 空间。

表 2 计算量对比(由 44.1 kHz 转换为 48 kHz)

算法	加法数/(次·秒 <sup>-1</sup> )	乘法数/(次·秒 <sup>-1</sup> )
FIR 算法	$1.537 \times 10^8$	$7.682 \times 10^7$
Farrow 算法	$2.945 \times 10^6$	$2.596 \times 10^6$
CIC 算法	$7.770 \times 10^5$	$1.803 \times 10^5$

表 3 存储空间对比( $N = 160, M = 147$ )

算法	ROM(Word)	RAM(Word)
FIR 算法	1 601	3 201
Farrow 算法	432	510
CIC 算法	0	24

由表 2、表 3 可见,CIC 算法在计算开销方面相较于 FIR 算法和 Farrow 算法具有极大的优势。

## 5 结 语

基于三级 CIC 滤波器原理的音频信号快速重采样

算法具有计算速度快、占用存储空间少、输出音质较好的特点,特别适合于在计算资源少、低成本的嵌入式音频播放系统中以软件形式实现,具有很高的性价比。当然,该算法仍有一定改进空间。将 CIC 滤波器级数升级为四级,将在增加少量计算开销的情况下进一步提高输出音频的质量。

## 参 考 文 献

- [1] 奥本海姆,谢弗. 离散时间信号处理[M]. 北京:电子工业出版社,2015.
- [2] Bregovic R, Yu Y J, Saramaki T, et al. Implementation of Linear-Phase FIR Filters for a Rational Sampling-Rate Conversion Utilizing the Coefficient Symmetry[J]. IEEE Transactions on Circuits and Systems, 2011,58(3): 548-561.
- [3] Kumar A, Yadav S, Purohit N. Generalized Rational Sampling Rate Conversion Polyphase FIR Filter[J]. IEEE Signal Processing Letters, 2017, 24(11): 1591-1595.
- [4] Rajamani K, Lai Y S, Farrow C W. An Efficient Algorithm for Sample Rate Conversion from CD to DAT[J]. IEEE Signal Processing Letters, 2000, 7(10): 288-290.
- [5] Chinaev A, Enzner G, Schmalenstroeer J. Fast and Accurate Audio Resampling for Acoustic Sensor Networks by Polyphase-Farrow Filters with FFT Realization[C]//Proceedings of ITG Fachtagung Sprachkommunikation(Speech Communications), Germany, Oldenburg, Oct. 2018: 96-100.
- [6] Chinaev A, Thune P, Enzner G. Low-Rate Farrow Structure with Discrete-Lowpass and Polynomial Support for Audio Resampling [C]//2018 26th European Signal Processing Conference(EUSIPCO), 2018: 475-479.
- [7] Abbas M, Gustafsson O, Johansson H. On the Fixed-Point Implementation of Fractional-Delay Filters Based on the Farrow Structure[J]. IEEE Transactions on Circuits and Systems, 2013, 60(4): 926-937.
- [8] Matthew P D. CIC filter introduction[EB/OL]. 2010-3-23. <http://home.mit.bme.hu/~kollar/papers/cic.pdf>.
- [9] 李凯勇. CIC 滤波器改进及其 FPGA 实现[J]. 现代电子技术, 2013, 36(1): 61-63.
- [10] Stosic B P, Pavlovic V D. Design of New Selective CIC Filter Functions with Passband-Droop Compensation[J]. Electronics Letters, 2016, 52(2): 115-117.
- [11] Romero D E T. Efficient CIC-Based Architecture with Improved Aliasing Rejection and Reduced Computational Complexity[J]. Electronics Letters, 2016, 52(15): 1294-1295.
- [12] 于进强,陶小鱼,欧斌. 基于 CIC 滤波的重采样技术[J]. 信息技术, 2008(12): 65-67.