

一种内插倍数可变的成型滤波器的 FPGA 设计实现

刘建梁¹ 孔 雅¹
LIU Jianliang KONG Ya

摘 要

在现代数字通信系统中, 基带成型滤波器是压缩信号频谱、抑制码间串扰的关键模块, 其性能直接影响系统传输质量。针对多码率场景下滤波器灵活性与实时性的需求, 文章提出了一种基于多相结构的内插倍数可变基带成型滤波器设计方案。详细介绍了升余弦成型滤波器的原理及系数计算过程, 着重阐述了 FIR 滤波器多相结构及 FPGA 实现方法。设计仿真结果表明, 该设计实现简单、逻辑资源消耗少、处理速度快、灵活性高, 可以较好地完成多码率基带信号的成型工作。

关键词

成型滤波器; 升余弦; 多相结构; FPGA

doi: 10.3969/j.issn.1672-9528.2025.09.022

0 引言

在现代无线数字通信系统中, 传输的信息量比较大, 占用频谱范围比较宽, 为了有效利用信道, 需要对基带信号进行频谱压缩; 另一方面, 不理想的系统传输总特性, 会产生码间串扰 (ISI), 导致前后码元的波形畸变、展宽和叠加, 造成误码率上升^[1]。升余弦成型滤波器可以对基带信号的频谱进行处理加工, 使其在消除码间干扰的同时, 抑制频谱带外扩散, 提高频谱的利用率^[2]。升余弦成型滤波器在实现上, 常规的做法是通过传统的乘累加方法来实现, 即输入信号与单位冲激响应进行线性卷积。随着滤波器长度增加, 这种方式会导致延迟增大, 资源紧张。本文从多相插值滤波器角度出发, 使用多相分解技术, 将原始成型滤波器分解为多个子滤波器, 每个时钟周期内只有一个子滤波器参与卷积运算, 大幅度降低了运算量。此外, 将码元信号幅度设置成 1, 避免了卷积乘法运算, 进一步降低了运算量, 提高了运行速度。同时借助滤波系数的可变步长寻址结构, 在 FPGA 系统时钟固定为 80 MHz 时, 实现了 625 kbit/s、1.25 Mbit/s、2.5 Mbit/s、5 Mbit/s、10 Mbit/s 和 20 Mbit/s 共 6 种码率的基带成型工作。

1 升余弦成型滤波器原理

采用合适的滤波器对码流进行滤波生成特定基带信号的过程称为“基带成型滤波”。根据奈奎斯特第一准则, 即使信号经传输后的波形发生了变化, 只要其特定点的采样值保持不变, 那么用再次采样的方法, 仍然可以准确地恢复原始信号。满足奈奎斯特第一准则的滤波器有很多种, 最简单的就是理想低通滤波器, 但是这种理想的低通滤波器是物理不

可实现的, 因为实际的滤波器不可能做到垂直截止, 并且时域拖尾过长, 导致计算复杂且容易产生偏差。在实际中得到广泛应用的是幅频响应, 具有奇对称升余弦状过渡带的一类滤波器, 即升余弦成型滤波器, 它有两个明显的优势: 一是有平滑的过渡带, 可以降低设计实现的难度; 二是通过引入滚降系数, 可以在定时同步误差和频谱宽度之间折中。理想的升余弦成型滤波器的冲激响应为:

$$h(n) = \frac{\sin(\pi n/T_s)}{\pi n/T_s} \cdot \frac{\cos(\alpha \pi n)/T_s}{1 - (4\alpha^2 n^2)/T_s^2} \quad (1)$$

式中: T_s 为码元周期; α 为滚降系数, 且 $0 \leq \alpha \leq 1$ 。滚降系数越小, 波形收敛越快, 对接收端定时同步要求越高; 滚降系数越大, 频谱在截止频率处越光滑, 频谱的利用率也越低。在实际应用中, 滚降系数的选取需要综合考虑频谱效率、信道特性、定时同步性能和抗干扰能力等多个因素。

令 $x(n)$ 为成型滤波器的输入码元序列, 当 $x(n)$ 经过冲激响应为 $h(n)$, 长度为 N 的成型滤波器时, 所对应的输出响应 $y(n)$ 为:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (2)$$

2 滤波器系数生成

本设计使用当前 MATLAB 工具推荐的 `rcosdesign` 函数进行升余弦滤波器的系数设计工作。函数原型为:

$$b = \text{rcosdesign}(\text{beta}, \text{span}, \text{sps}, \text{shape}) \quad (3)$$

式中: beta 为滚降系数, 取值 $0 \sim 1$; span 为滤波器截断的符号长度, 即冲激响应持续的符号数, 实际设计中一般在 $8 \sim 16$ 之间取值; sps 为单个符号范围的采样个数, 即过采样倍数; shape 为生成的滤波器类型。参数为“norm”时返回升余成

1. 天津讯联科技有限公司 天津 300358

型弦滤波器系数，参数为“sqrt”时返回平方根升余弦成型滤波器系数。

本方案滚降系数 $\beta=0.35$ ，截断符号长度 $\text{span}=8$ ，成型滤波器类型 $\text{shape} = \text{'normal'}$ 。为实现 625 kbit/s~20 Mbit/s 多码率的成型要求，MATLAB 滤波器设计时应选用最高过采样倍数（最低码率），因此 $\text{sps}=F_s/R_b=80 \times 10^6/625\,000=128$ ，最终生成了 1 025 个滤波系数，并把系数量化成 8 位有符号整数，量化后的时域响应波形如图 1 所示。

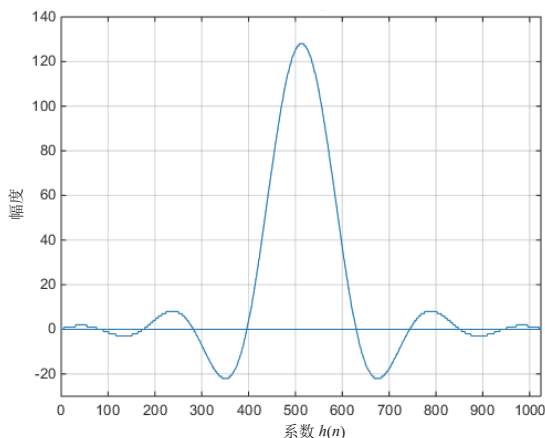


图 1 滤波器时域响应波形

多相 FIR 滤波器的分解对滤波器的长度 N 和内插倍数 M 的关系进行了限制，要求长度 N 应是内插倍数 M 的整数倍。需要说明的是，这个倍数要求只是为了便于描述多相分解的实现结构，并没有对 MATLAB 生成的滤波器长度进行约束，如果不满足倍数关系，可以通过补零系数或者舍弃系数来修正。由于 rcosdesign 函数生成的升余弦滤波器长度 $N=\text{span} \times \text{sps}+1$ ，与内插倍数不构成整数倍的关系，因此需要对系数进行优化处理。文献 [3-4] 中采用舍弃中心系数再平移的方式，处理后的系数满足第一类线性相位，虽然可以利用系数的对称性简化后续设计，但会导致滤波器输出波形的发生变化。观察到经过 8 位量化后， $h(0)=h(1\,024)=0$ ，因此本设计采用直接舍弃系数 $h(1\,024)$ 的方法，这种方式不会对波形精度造成影响。最终滤波器长度 $N=1\,024$ ，系数范围为 $h(0) \sim h(1\,023)$ 。如果需要增大量化位宽，此时 $h(1\,024)$ 可能不再为零，则应采用补零系数的方式来满足倍数关系。

3 多相结构设计

多相分解（polyphase decomposition），又称为多相表示，是多速率信号处理的一种有效方法，是指将 FIR 数字滤波器的转移函数 $H(z)$ 分解成若干个不同的相位成分，从而达到提高计算效率的目的^[5]。已知长度为 N 的 FIR 滤波器，按第一型进行分解成 M 组：

$$H(z) = \sum_{k=0}^{M-1} z^{-k} E_k(z^M) \quad (4)$$

$$\text{其中: } E_k(z) = \sum_n h[Mn+k]z^{-n} \quad (5)$$

式中：为 $H(z)$ 的第 k 个多相成分。

第一型多相分解的结构如图 2 所示。

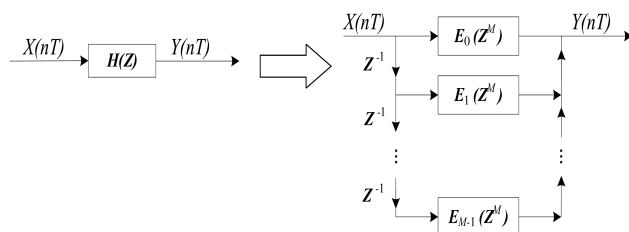


图 2 第一型多相分解结构图

对于滤波器长度 $N=1\,024$ ，内插倍数为 $M=128$ 的内插操作，波器被分成 M 个子滤波器，每个子滤波器长度 $Q=N/M=8$ ，第一个子滤波器系数为 $h(n)$ 序号中是 M 的整数倍的样点，第二个子滤波器系数为 $h(n)$ 序号中是 M 整数倍加 1 的样点，以此类推，各子滤波器系数如下：

$$h_0^{(E)} = [h(0), h(128), h(256), h(384), h(512), h(640), h(768), h(896)]$$

$$h_1^{(E)} = [h(1), h(129), h(257), h(385), h(513), h(641), h(769), h(897)]$$

$$h_2^{(E)} = [h(2), h(130), h(258), h(386), h(514), h(642), h(770), h(898)]$$

.....

$$h_{126}^{(E)} = [h(126), h(254), h(382), h(510), h(638), h(766), h(894), h(1\,022)]$$

$$h_{127}^{(E)} = [h(127), h(255), h(383), h(511), h(639), h(767), h(895), h(1\,023)]$$

内插滤波器的多相分解，本质上是利用相邻码元间内插了 $M-1$ 个 0，而这些内插的 0 对相乘求和结果没有任何贡献，因此只需要计算与实际码元相关的部分即可。令 $x(n)$ 为输入码元序列， $y(n)$ 为滤波器的输出响应，经过多相滤波后，用公式表示为：

$$y(n) = \sum_{k=0}^{Q-1} h_w^E(k)x(n-k) \quad 0 \leq w < M \quad (6)$$

当 $M=128$ 时，每个时钟周期滤波运算表达式如表 1 所示。

表 1 $M=128$ 时每个时钟滤波运算

时钟	滤波运算表达式
clk0	$x(0)h(896)+x(1)h(768)+x(2)h(640)+x(3)h(512)+x(4)h(384)+x(5)h(256)+x(6)h(128)+x(7)h(0)$
clk1	$x(0)h(897)+x(1)h(769)+x(2)h(641)+x(3)h(513)+x(4)h(385)+x(5)h(257)+x(6)h(129)+x(7)h(1)$
.....	
clk127	$x(0)h(1023)+x(1)h(895)+x(2)h(767)+x(3)h(639)+x(4)h(511)+x(5)h(383)+x(6)h(255)+x(7)h(127)$

表 1 (续)

时钟	滤波运算表达式
clk128	$x(1)h(896)+x(2)h(768)+x(3)h(640)+x(4)h(512)+$ $x(5)h(384)+x(6)h(256)+x(7)h(128)+x(8)h(0)$
.....	

4 多相结构的 FPGA 实现

多相结构 FPGA 设计由四部分组成：码型映射单元、可变寻址单元、子滤波器系数 ROM 表、滤波计算单元。多相结构设计总体框图如图 3 所示。

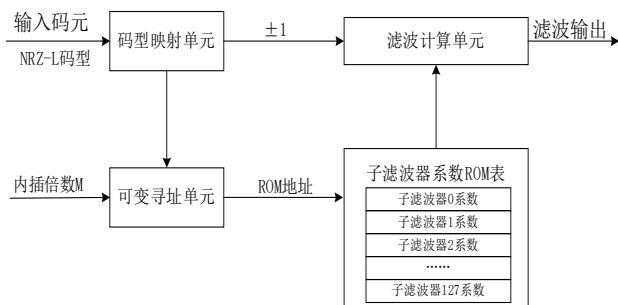


图 3 多相结构总体框图

(1) 码型映射单元

该模块把输入的 NRZ-L 码型的码元进行 2 bit 映射，映射关系为 $b=1-2a$ ，其中 a 为输入码元， b 为输出符号。即输入码元 0 映射成“01”，输入码元 1 映射成“11”。

(2) 可变寻址单元

该模块用于控制系数 ROM 的地址，根据不同的内插倍数 M ，调整步长，循环产生 ROM 表的地址。在每个码元周期内，码元不动，每个时钟周期子滤波器地址会步进一次，直至遍历完所有的有效地址，从而实现内插倍数可变。在 FPGA 实现步长调整时，可以直接通过对 ROM 地址截位来完成，伪代码如下：

```
case (M)
    128: coe_addr = rom_addr[6:0]; // 步长 1
    64: coe_addr = rom_addr[6:1]; // 步长 2
    32: coe_addr = rom_addr[6:2]; // 步长 4
    16: coe_addr = rom_addr[6:3]; // 步长 8
    8: coe_addr = rom_addr[6:4]; // 步长 16
    4: coe_addr = rom_addr[6:5]; // 步长 32
endcase
```

(3) 子滤波器系数 ROM 表

该模块用于存放子滤波器系数。二维 ROM 表每行位宽 64 位，内容为每个子滤波器的 8 个量化后系数，ROM 表深度为 128，代表有 128 组子滤波器系数 ($h_0^{(i)} \sim h_{127}^{(i)}$)，总共有 1 024 个系数。

(4) 滤波计算单元

该模块实现了滤波相关的计算操作。由于输入码元被映射成了 ± 1 ，所以不需要乘法操作，如果映射后符号为 $+1$ ，则乘法计算结果为系数本身；如果映射后符号是 -1 ，则乘法计算结果为系数取反加 1。然后再对 8 个结果进行两级流水相加，就可以得到成形滤波输出^[6]。

在码率为 625 kbit/s（内插 128 倍）时的成型滤波器仿真波形如图 4 所示。

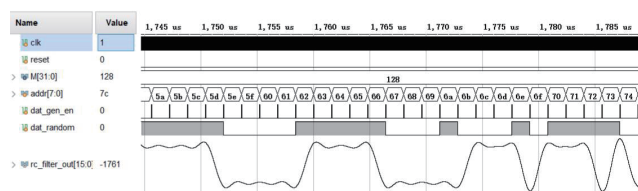


图 4 成型滤波器仿真波形

5 结语

本文根据多相分解相关理论，提出了一种内插倍数可变的成型滤波器的设计方法。该方法极大地减少了滤波器的运算规模，节省了逻辑资源，提高了运行速度。同时结合滤波器系数 ROM 表地址结构，实现了码元内插倍数可变。该方法也可以推广到其他形式 FIR 内插滤波器的实现。

参考文献：

- [1] 樊昌信, 曹丽娜. 通信原理 [M]. 北京: 国防工业出版社, 2012.
- [2] 梁尧, 周智勋, 何丽. 一种基于查找表的 FIR 成型滤波实现装置 [J]. 太赫兹科学与电子信息学报, 2015, 13(1): 102-105.
- [3] 赵林军. 根升余弦脉冲成形滤波器 FPGA 实现 [J]. 现代电子技术, 2011, 34(1): 23-28.
- [4] 李艳萍, 周胜源. 高速成形滤波器的研究与实现 [J]. 电子设计工程, 2012, 20(8): 103-105.
- [5] 陶然, 石岩, 王越. 多抽样率数字信号处理理论及其应用 (第 2 版) [M]. 北京: 清华大学出版社, 2022.
- [6] 单琦, 张志芳, 沙立伟, 等. 高速并行成形滤波器的 FPGA 实现方法 [J]. 电子设计工程, 2018, 26(3): 89-92.

【作者简介】

刘建梁 (1989—)，男，辽宁庄河人，硕士，工程师，研究方向：遥控遥测、高速总线。

孔雅 (1988—)，男，湖北通城人，硕士，工程师，研究方向：遥控遥测、自组网通信。

(收稿日期：2025-04-27 修回日期：2025-09-15)