

저전력 Uniform 필터뱅크 구현을 위한 블록 필터 아키텍처

양 세 정, 장 영 범
이화여자대학교 정보통신학과
전화 : 02-3277-4251 / 핸드폰 : 011-445-2830

Block Filter Architecture for Low-power Uniform Filter Banks Implementation

Se Jung Yang, Young-Beom Jang
Dept. of Information Electronics Engineering, Ewha Womans University
E-mail : ybjang@mm.ewha.ac.kr

Abstract

Block filter implementation technique for uniform filter banks is proposed in this paper. By applying block filters into decimation and interpolation filters, it is shown that down and up samplers are cancelled out in respective filters. Furthermore, by applying block filters into uniform filter banks, significant reduction for computational complexity is achieved since prototype filter can be shared in each channel implementation. Also, it is shown that proposed implementation is a reconfigurable structure in terms of filter order variation.

I. 서론

필터뱅크를 사용하는 통신 시스템에서는 필터뱅크 계산의 복잡도와 유연성이 동시에 고려되어 연구되고 있다. 필터뱅크의 유연성이란 필터의 차수가 변화하는 경우에 효과적으로 적용할 수 있는 구조를 의미한다. 기존의 Uniform DFT 필터뱅크의 경우에 Polyphase 구조가 가장 많이 사용되는데, 그 까닭은 첫 번째 채널에 사용되는 필터(Prototype 필터라고 함)가 다른 모든 채널에서도 공용으로 사용되어, 계산량 감소를 통한 저전력 구현이 가능하기 때문이다. 또한 DFT의 구현도 FFT를 사용하여 역시 계산량을 줄일 수 있다.[1] 그런데, 필터뱅크가 응용되는 다중채널 DDC(Digital Down Converters)와 DUC(Digital Up Converters)에서는 필터뱅크의 저전력 구현과 더불어 필터의 유연성이 특히 요구된다. 본 논문은 이와 같은 계산량의 감소와 유연성까지

갖춘 필터뱅크를 블록 필터 구조를 사용하여 효율적으로 구현하는 방법을 제안한다.

II. 데시메이션과 인터플레이션의 블록 필터 구현

2.1 블록 Direct form 구조의 데시메이션

블록 필터 구조는 병렬 처리를 통한 필터의 연산 속도를 높이기 위하여 개발되었다.[2][3] 이 절에서는 데시메이션 필터를 블록필터구조로 구성함으로써 뒤에 붙는 데시메이터와의 상쇄가 발생하여 구조를 간략하게 하는 방법을 제안한다. 그림 1(a)와 같은 데시메이션 필터와 데시메이터에 블록 Direct form 필터를 적용하면 그림 1(b)와 같게 되며, 블록 필터는 SPC (Serial to Parallel Converters, 직렬-병렬 변환기), 블록 필터, 그리고 PSC(Parallel to Serial Converters, 병렬-직렬 변환기)로 구성된다. 이 때, 블록 필터의 입력벡터의 크기는 뒤에 붙는 데시메이터의 인수의 크기와 같도록 해야 한다. 그림 1(b)에서 점선의 원으로 둘러싸인 PSC와 데시메이터는 등가적으로 그림 1(c)와 같이 표현될 수 있다. 즉, PSC는 병렬 신호들을 Expander에 의해 제로들을 삽입시키고, 지연소자들을 사용하여 차례로 더함으로써 그림 1(c)의 $p[n]$ 의 신호를 얻을 수 있다. 그림 1(c)의 k 번째 Expander 출력은 다음과 같이 표현된다.

$$g_k[n] = \begin{cases} f_k[n/L], & n=0, L, 2L, \dots \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

따라서 $p[n]$ 은 다음과 같이 나타낼 수 있다.

$$p[n] = g_0[n] + g_1[n-1] + \dots + g_{L-1}[n-L+1] \quad (2)$$

$$= \begin{cases} f_0[\frac{n}{L}], & n=0, L, 2L, \dots \\ f_1[\frac{n-1}{L}], & n=1, L+1, 2L+1, \dots \\ \vdots \\ f_{L-1}[\frac{n-(L-1)}{L}], & n=L-1, 2L-1, \dots \end{cases}$$

데시메이터는 두 단의 동작으로 나누어 표현된다. 첫 단에서는 L 의 정수 배를 제외한 모든 점을 0으로 바꾸는 동작이므로 다음과 같이 나타내진다.

$$p'[n] = \begin{cases} f_0[\frac{n}{L}], & n=0, L, 2L, \dots \\ 0, & otherwise \end{cases} \quad (3)$$

두 번째 단에서는 $p'[n]$ 이 시간 축에서 압축되어 0들이 사라지므로 최종적으로 데시메이터의 출력은 다음과 같이 표현될 수 있다.

$$q[n] = f_0[n], \quad n=0, 1, 2, \dots \quad (4)$$

이 식에서 보듯이 데시메이터의 출력은 PSC의 병렬 입력 중에서 맨 위의 신호가 됨을 알 수 있다. 따라서 블록 필터의 병렬 출력 중에서 맨 위의 출력만이 필요하므로 그림 1(d)와 같이 행렬 계수들을 벡터 계수로 간략화할 수 있다. 본 논문에서는 블록 필터 구조로서 블록 Direct form을 선택하였으며 블록의 크기를 L 로 정의함으로써 블록 시스템 함수는 다음과 같이 나타낼 수 있다.

$$H_0(z) = H_{00} + H_{01}z^{-1} + \dots + H_{0M}z^{-M} \quad (5)$$

이 식에서 행렬 계수들은 다음과 같다.

$$H_{00} = \begin{bmatrix} h_0 & 0 & \dots & 0 \\ h_1 & h_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{L-1} & h_{L-2} & \dots & h_0 \end{bmatrix}, \quad H_{01} = \begin{bmatrix} h_L & h_{L-1} & \dots & h_1 \\ h_{L+1} & h_L & \dots & h_2 \\ \vdots & \vdots & \ddots & \vdots \\ h_{2L-1} & h_{2L-2} & \dots & h_L \end{bmatrix} \quad (6)$$

$$\dots, \quad H_{0M} = \begin{bmatrix} h_N & h_{N-1} & \dots & h_{N-L+1} \\ 0 & h_N & \dots & h_{N-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h_N \end{bmatrix}$$

식(6)에서 N 은 주어진 데시메이션 필터의 차수이며, M 은 블록 필터의 차수이다. 여기서 M 은 N/L 보다 크거나 같은 최소의 정수로 정의된다. (6)의 행렬계수를 사용한 블록 Direct form 구조가 그림 1(b)이다. 앞에서 보인 PSC와 데시메이터의 관계를 이용한 그림 1(d)의 벡터 계수들은 다음과 같다.

$$\begin{aligned} H_{00} &= [h_0 \ 0 \ \dots \ 0], \\ H_{01} &= [h_L \ h_{L-1} \ \dots \ h_1], \\ &\vdots \\ H_{0M} &= [h_N \ h_{N-1} \ \dots \ h_{N-L+1}]. \end{aligned} \quad (7)$$

위의 벡터계수들은 식(6)의 행렬계수들의 첫째 행과 같으며 그림 1(d)와 같은 간단한 구조의 데시메이션을 얻을 수 있었다. 즉, PSC와 데시메이터의 상쇄와 더불어 행렬계수가 벡터

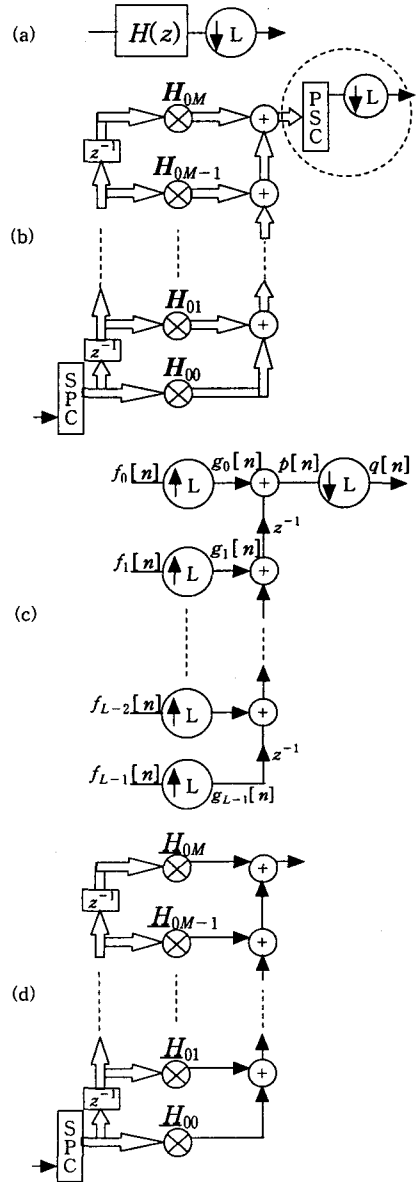


그림 1. 블록필터를 사용한 데시메이션 구조
(a)데시메이션, (b)블록필터와 데시메이터, (c)PSC와 데시메이터, (d)최종의 블록필터를 사용한 데시메이션

계수로 간략화되는 구조를 얻을 수 있다.

2.2 블록 Direct form 구조의 인터플레이션

이 절에서는 블록필터를 인터플레이션에 적용하는 방법을 제안한다. 블록 시스템 함수는 다음과 같다.

저전력 Uniform 필터 뱅크 구현을 위한 블록 필터 아키텍처

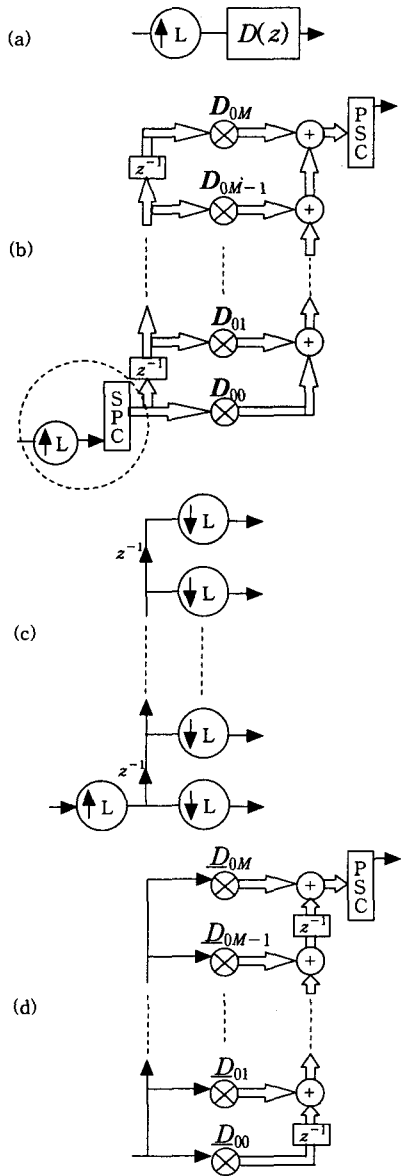


그림 2. 블록필터를 사용한 인터플레이션 구조
 (a)인터플레이션, (b)블록필터를 사용한 인터플레이션,
 (c)Expander와 SPC, (d)최종의 블록필터를 사용한
 인터플레이션

$$D_0(z) = D_{00} + D_{01}z^{-1} + \dots + D_{0M}z^{-M} \quad (8)$$

식(8)의 블록 시스템 함수에서 행렬 계수는 데시메이터에서와 같은 방법으로 유도되므로 다음과 같이 나타낼 수 있다.

$$D_{00} = \begin{bmatrix} d_0 & 0 & \dots & 0 \\ d_1 & d_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ d_{L-1} & d_{L-2} & \dots & d_0 \end{bmatrix}, D_{01} = \begin{bmatrix} d_L & d_{L-1} & \dots & d_1 \\ d_{L+1} & d_L & \dots & d_2 \\ \vdots & \vdots & \ddots & \vdots \\ d_{2L-1} & d_{2L-2} & \dots & d_L \end{bmatrix} \quad (9)$$

$$\dots, D_{0M} = \begin{bmatrix} d_N & d_{N-1} & \dots & d_{N-L+1} \\ 0 & d_N & \dots & d_{N-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & d_N \end{bmatrix}$$

식(9)에서의 N, M, L 은 데시메이션 필터 경우와 같이 정의된다. 따라서 그림 2(b) 구조의 계수들이 식(9)로 표현된다. 이 구조에 데시메이터에서와 같은 방법을 사용하여 Expander와 SPC를 상쇄시키면 그림 2(d)의 효율적인 구조를 얻게 되며 벡터계수는 다음과 같다.

$$\begin{aligned} D'_{00} &= [d_0 \quad d_1 \quad \dots \quad d_{L-1}], \\ D'_{01} &= [d_L \quad d_{L+1} \quad \dots \quad d_{2L-1}], \\ &\vdots \\ D'_{0M} &= [d_N \quad 0 \quad \dots \quad 0]. \end{aligned} \quad (10)$$

III. 필터 뱅크의 블록 필터 구조

필터 뱅크의 분석 단에는 채널 수만큼의 데시메이션 필터가 사용된다. 0부터 $L-1$ 까지 L 개의 채널을 갖는 분석 단의 구조를 제안한다. Uniform 필터 뱅크의 경우에 l 번째 채널의 데시메이션 필터는 prototype 필터의 shift 버전이므로 다음과 같이 나타내진다.

$$\begin{aligned} H_l(z) &= H_0(z e^{j\frac{2\pi}{L}l}) \\ &= h_0 + h_1 z^{-1} e^{j\frac{2\pi}{L}l} + h_2 z^{-2} e^{j\frac{2\pi}{L}2l} + \dots + h_N z^{-N} e^{j\frac{2\pi}{L}Nl} \end{aligned} \quad (11)$$

이 시스템 함수로부터 식(5)와 같은 블록 시스템 함수가 쉽게 유도될 수 있다.[3] 이와 같이 유도된 블록 시스템 함수의 행렬 계수들로부터 첫째 행들만을 뽑아서 벡터 계수가 얻어지며, 다음과 같이 나타내진다.

$$\begin{aligned} H_{l0} &= [h_0 \quad 0 \quad \dots \quad 0 \quad 1], \\ H_{l1} &= [h_L e^{j\frac{2\pi}{L}Ll} \quad h_{L-1} e^{j\frac{2\pi}{L}(L-1)l} \quad \dots \quad h_1 e^{j\frac{2\pi}{L}l}], \\ &\vdots \\ H_{lM} &= [h_N e^{j\frac{2\pi}{L}Nl} \quad h_{N-1} e^{j\frac{2\pi}{L}(N-1)l} \quad \dots \quad h_{N-L+1} e^{j\frac{2\pi}{L}(N-L+1)l}]. \end{aligned} \quad (12)$$

식(12)를 좀더 간단히 표현하기 위해서 $W_L = e^{-j\frac{2\pi}{L}}$ 를 정의하면 식(12) 벡터들의 두 번째 엘리먼트들은 모두 W_L^l 을 사용하여 나타낼 수 있다. 예를 들면 H_{l1} 의 두 번째 엘리먼트는 다음과 같이 표현될 수 있다. 즉,

$$h_{L-1} e^{j\frac{2\pi}{L}(L-1)l} = h_{L-1} e^{-j\frac{2\pi}{L}l} = h_{L-1} W_L^l. \quad (13)$$

이와 같이 같은 번째의 엘리먼트들은 같은 W_L^k 를 사용하여 나타낼 수 있다. 또한, 위의 벡터들의 마지막 엘리먼트들은 다음과 같이 표현할 수 있다.

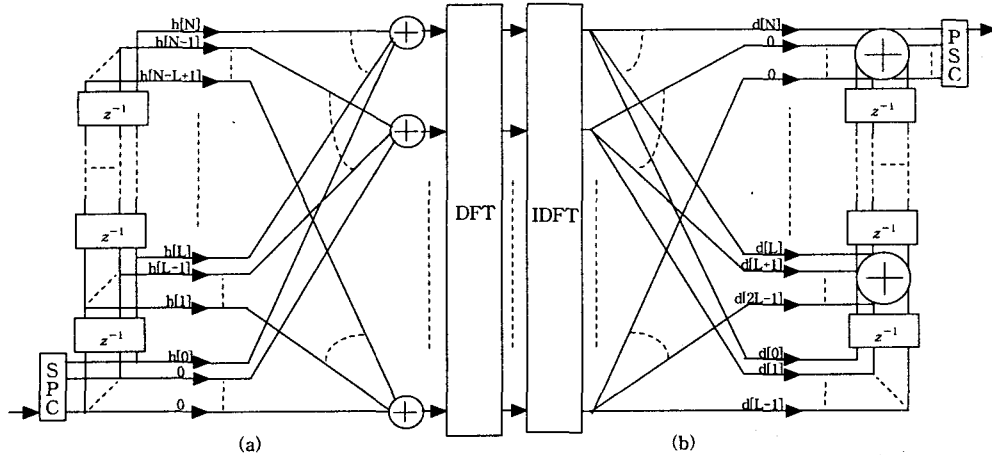


그림 3. Uniform 필터 बैं크의 블록 필터 구조, (a)분석단, (b)합성단

$$h_1 e^{j\frac{2\pi}{L}l} = h_1 e^{j\frac{2\pi}{L}l} e^{-j2\pi l} = h_1 e^{-j\frac{2\pi}{L}l(L-1)} = h_1 W_L^{l(L-1)}. \quad (14)$$

따라서 식(12)의 벡터 계수들은 W_L^k 를 사용하여 다음과 같이 표기할 수 있다.

$$\begin{aligned} H_{l0} &= [h_0 \quad 0 \quad \dots \quad 0], \\ H_{l1} &= [h_L \quad h_{L-1} W_L^l \quad \dots \quad h_1 W_L^{l(L-1)}], \\ &\vdots \\ H_{lM} &= [h_N \quad h_{N-1} W_L^l \quad \dots \quad h_{N-L+1} W_L^{l(L-1)}]. \end{aligned} \quad (15)$$

식(7)과 (15)를 비교해보면, 식(15)에서 W_L^k 들만 제거하면 식(7)과 같기 때문에 W_L^k 들만을 분리하여 구현하면 식(7)의 필터를 공유할 수 있게 된다. 즉, 첫 번째 채널의 필터를 l 번째 채널에서도 공유할 수 있음을 의미한다. 모든 채널에서 분리된 W_L^k 들의 구현은 DFT가 된다. 전체적인 분석단의 블록 필터 구조는 그림 3(a)와 같다.

이변에는 필터 बैं크의 합성단에 사용되는 l 번째 인터폴레이션 필터를 고려해보기로 하자. 역시 Uniform 필터 बैं크이므로 다음과 같이 prototype 필터를 shift 하여 표현할 수 있다.

$$\begin{aligned} D_l(z) &= D_0(z e^{j\frac{2\pi}{L}l}) \\ &= d_0 + d_1 z^{-1} e^{j\frac{2\pi}{L}l} + d_2 z^{-2} e^{j\frac{2\pi}{L}2l} + \dots + d_N z^{-N} e^{j\frac{2\pi}{L}Nl}, \end{aligned} \quad (16)$$

분석 단과 같은 방법으로, 위의 시스템 함수로부터 다음과 같은 벡터 계수들을 유도하였다.

$$\begin{aligned} D'_{l0} &= [d_0 \quad d_1 W_L^{-l} \quad \dots \quad d_{L-1} W_L^{-l(L-1)}], \\ D'_{l1} &= [d_L \quad d_{L+1} W_L^{-l} \quad \dots \quad d_{2L-1} W_L^{-l(L-1)}], \\ &\vdots \\ D'_{lM} &= [d_N \quad 0 \quad \dots \quad 0]. \end{aligned} \quad (17)$$

식(17)에서도 역시 W_L^k 들을 제거하면 식(10)과 같아지므로

W_L^k 들을 분리하면 식(10)의 필터를 l 번째 채널에서 공유할 수 있으며, 이는 모든 채널에서 공유할 수 있음을 의미한다. 합성 단의 그림 3(b)에서 보듯이 분리된 W_L^k 들의 구현은 IDFT가 된다.

IV. 결론

본 논문은 블록 필터를 사용하여 효율적인 필터 बैं크 구조를 제안하였다. 블록 필터를 데시메이션 필터나 인터폴레이션 필터에 적용할 경우 상쇄에 의하여 계산량이 감소됨을 보였다. 또한 블록 필터를 Uniform 필터 बैं크에 적용할 경우에는 첫째 채널의 필터가 모든 채널에서 공유할 수 있어 계산량이 감소됨을 보였다. 따라서 제안된 필터 बैं크는 계산량이 감소를 통한 저전력 구현이 가능하다. 또한 필터의 차수가 변화하더라도 필터 बैं크의 블록 추가로 유연하게 적용하는 구조를 얻을 수 있었다.

참고문헌

[1] M. Bellanger, G. Bonnerot, and M. Coudreuse, "Digital filtering by polyphase network: Application to sample rate alteration and filter banks," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-24, pp. 109-114, Apr. 1976.
 [2] C. W. Barnes, and S. Shinnaka, "Block-Shift Invariance and Block Implementation of Discrete-Time Filters," IEEE Trans. Circuits and Systems, vol. CAS-27, No. 8, pp. 667-672, Aug. 1980.
 [3] Y. Jang, and S. P. Kim, "Block digital filter structures and their finite precision responses," IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing, vol. 43, No. 7, pp. 495-506, July 1996.