

디지털 IF 다운 샘플러와 업 샘플러의 저전력 블록 필터링 아키텍처

정회원 장영범*, 김낙명*

Low-Power Block Filtering Architecture for Digital IF Down Sampler and Up Sampler

Young-Beom Jang*, Nak-Myeong Kim* *Regular Members*

요약

본 논문에서는, 디지털 IF 다운 샘플러와 업 샘플러의 저전력 구현을 위한 블록 필터링 아키텍처를 제안한다. 소프트웨어 라디오와 같은 차세대 이동통신 방식에서 디지털 IF 다운 샘플러와 업 샘플러의 효율적인 구현방법이 더욱더 요구되어가고 있는 추세이다. 디지털 IF 다운 샘플러는 앞단에 데시메이션 필터를 수반하여 구성되며, 업 샘플러는 뒷단에 인터플레이션 필터가 수반되어 구성된다. 본 논문의 다운 샘플러 아키텍처에서는 블록 필터링 구조가 갖는 병렬처리 구조를 이용하여 필터를 구현하였으며, 블록 필터링 아키텍처에서 구조적으로 생겨나는 업 샘플링을 직렬로 연결되는 다운 샘플러와 상쇄시킴으로서 효과적인 구조가 만들어짐을 보인다. 또한 업 샘플러 아키텍처에서는 인터플레이션 필터의 블록 필터링에서 만들어지는 다운 샘플링이 앞단의 업 샘플러와 상쇄되어 병렬처리와 저속의 처리가 가능해짐을 보인다. 본 논문에서 제안된 아키텍처와 Polyphase 아키텍처를 비교 분석한다.

ABSTRACT

In this paper, low-power block filtering architecture for digital IF down sampler and up sampler is proposed. Software radio technology requires low power and cost effective digital IF down and up sampler. Digital IF down sampler and up sampler are accompanied with decimation filter and interpolation filter, respectively. In the proposed down sampler architecture, it is shown that the parallel and low-speed processing architecture can be produced by cancellation of inherent up sampler of block filter and down sampler. Proposed up sampler also utilizes cancellation of up sampler and inherent down sampler of block filtering structure. The proposed architecture is compared with the conventional polyphase architecture.

I. 서론

이동 통신의 괄목할만한 발전속도로 말미암아, 휴대용 단말기는 저전력화와 고속 데이터 처리의 중요성이 점점 더 증대되고 있다. 이와 병행하여, 여러 가지의 표준에 따른 알고리즘들을 단말기를 교체하지 않고도 채용할 수 있는 소프트웨어 라디오

의 개념이 차세대 이동통신을 주도할 것이라는 주장이 설득력을 얻어가고 있다. 이렇게 되면 IF 처리를 디지털신호처리를 사용해야 여러 가지 표준의 서로 다른 다운 샘플러 및 업 샘플러들을 적응적으로 구현할 수 있게 될 것이다. 따라서 앞으로 디지털 IF는 적응성, 저전력의 구현, 그리고 고속프로세싱을 모두 요구하게 될 것이다. 특히 협대역 디지털 IF는 매우 100 이상의 인수로 데시메이션을 요구하

* 이화여자대학교 정보통신학과(ybjang@mm.ewha.ac.kr, nmkim@mm.ewha.ac.kr)

논문번호 : 00047-0207, 접수일자 : 2000년 2월 7일

* 본 연구는 교육부 BK21 지원에 의해 수행되었습니다.

고 있으며, 예로서 협대역 디지털 IF를 사용하는 GSM의 경우에 IF의 입력 대역폭은 25MHz이고 신호 대역폭은 200KHz를 사용하고 있다.

디지털 IF가 구현되는 하드웨어는 프로그래머블 DSP, Hard-wired DSP, 그리고 벡터 프로세서 등이 있을 수 있으며, 각각마다 최적의 구현 아키텍처가 개발될 수 있을 것이다. 그러나 이러한 구현 하드웨어가 결정되지 않은 상황에서 일반적인 아키텍처의 개발 요구가 증대되고 있다. 그림 1의 (a)는 다운 샘플러 (b)는 업 샘플러를 각각 나타내고 있다. (a)의 다운 샘플러는 전단의 필터와 결합되어 데시메이션을 수행하게되며, 이는 정수 인수인 L 만큼의 샘플링 속도를 낮추는 동작이다. 다운 샘플러는 원안에 아래방향의 화살표와 데시메이션 인수인 L 로서 표현되어 진다. (b)의 업 샘플러는 후단의 필터와 결합하여 인터플레이션을 수행하게된다. 이는 역시 마찬가지로 정수 인수인 L 만큼 샘플링 속도를 높이는 작업이다. 업 샘플러도 역시 원안에 윗방향의 화살표와 인터플레이션 인수인 L 로 표현한다. 그림 1.과 같은 샘플러가 사용되는 소프트웨어 라디오 개념의 휴대용 이동 단말기나 기지국의 디지털 IF가 가져야 하는 적응성으로서는 필터의 차수와 다운 샘플링과 업 샘플링의 크기가 변수가 될 것이다. 다운 샘플러를 생각해 보면, 여러 가지 표준의 다운 샘플러마다 필터의 차수와 다운 샘플링의 크기가 변화할 것이며, 이에 따른 적응적 알고리즘이나 아키텍처가 필요하게 된다. 어떤 표준에서 요구되는 다운 샘플링의 크기를 L이라고 할 때 L보다 작은 크기의 다운 샘플러 밖에 가용하지 않을 경우를 가정해 보자. 이 경우에는 IFIR(Interpolated Finite Impulse Response) 필터를 사용하여 필터를 직렬로 2개의 필터로 분할하고, 이를 다시 그림2.의 (b)와 같은 등가구조를 사용하여 필터와 다운 샘플링을 뒤바꿈으로서 2 스테이지의 다운 샘플러를 구성하는 방법이 제안되었다.^[1]

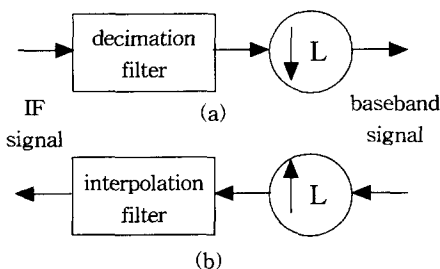


그림 1. (a)다운 샘플러, (b)업 샘플러

디지털 IF의 또 다른 요구조건인 저전력과 고속 처리를 위해서도 여러 가지 연구가 진행되고 있다. 현재의 상업적으로 구현 가능한 단말기의 반도체 집적기술로는 디지털 IF의 병렬처리가 불가피하다. 즉, 디지털 IF의 입력데이터속도를 기존의 필터 아키텍처로 구현하려면 매우 고속의 필터가 필요하게 되므로, 여러 가지의 병렬처리 필터구조가 제안되고 있다. 그중 대표적인 것으로서 Polyphase 아키텍처의 다운 샘플러가 있다. 이는 데시메이션 필터를 Polyphase로 분해하여 병렬로 처리함으로써 필터링해야하는 작업을 분산시켰으며, 다시 그림2. (b)의 등가구조를 사용하여 Polyphase 필터와 다운 샘플러의 위치를 뒤바꿈으로서 입력신호에 대한 처리클럭을 다운 샘플링의 배수만큼 낮추는 저전력 구조를 제안하고 있다.^[2] 이 구조에서는 병렬화된 이후의 각 노드들의 필터링은 원래의 필터 탭수에 비례한다. 예로서 64 탭의 필터를 4의 크기로 다운 샘플링하는 경우에, 각 병렬화된 노드의 필터 탭수는 16이 된다. 이는 16번의 곱셈을 필요로 하는 MAC(Multiplication And Accumulation)이 4개가 필요함을 의미한다. 이 경우에, 원래의 필터 탭수가 128로 바뀐다면 32번의 곱셈을 필요로 하는 MAC이 4개가 필요하게 된다. 즉, MAC의 크기가 고정되어있는 프로세서를 사용하는 경우에는 적응적으로 동작하기가 용이하지 않다.

본 논문은 벡터 프로세서와 같은 MAC의 크기가 고정된 엔진을 사용하는 경우에, 필터의 탭수의 변화에 적응하기 용이한 아키텍처를 제안한다. 블록 필터를 도입함으로써 병렬처리의 구조와 저속처리의 장점은 그대로 유지하면서, 원래의 필터 탭수가 늘어어나더라도 고정크기의 MAC 엔진의 수만 늘어나는 효율적인 아키텍처가 됨을 보인다.

논문의 구성은 II장에서는 기존의 병렬처리 아키텍처로 제안된 Polyphase를 사용한 다운 샘플러의 구조를 살펴본다. 업 샘플러의 경우도 동일한 방법으로 유도될 수 있으므로 이 유도과정은 생략하였다. III장에서는 본 논문에서 제안하는 블록 필터링을 사용하는 디지털 IF 다운 샘플러와 업 샘플러 아키텍처의 유도과정을 살펴본다. 그리고 IV장과 V장에서는 아키텍처의 비교분석 예제와 결론을 각각 기술한다.

II. 기존의 Polyphase 다운 샘플러 아키텍처

일반적인 필터의 Polyphase 분할기법은 [3]에서

제안되어졌다. Polyphase 기법은 필터의 병렬처리를 가능케 해주는 매우 유용한 구조이며, 특히 데시메이션과 인터플레이션과 같은, 다운 샘플링과 업 샘플링이 포함된 구조에서는 처리 클럭을 다운 샘플링이나 업 샘플링의 크기만큼 낮출 수 있으므로 더욱 유용한 아키텍처가 유도되어 진다. 이장에서는 이러한 Polyphase 다운 샘플러 아키텍처의 유도과정을 소개하기로 한다. 그림 1.의 (a)에서 앞단의 데시메이션 필터는 다음 식과 같이 Polyphase의 여러 개의 필터의 합으로 나타내어 질 수 있다.

$$\begin{aligned}
 H(z) &= h_0 + h_1 z^{-1} + h_2 z^{-2} + \dots + h_L z^{-L} + \dots \\
 &= (h_0 + h_L z^{-L} + h_{2L} z^{-2L} + \dots) \\
 &+ z^{-1} (h_1 + h_{L+1} z^{-L} + h_{2L+1} z^{-2L} + \dots) \\
 &+ z^{-2} (h_2 + h_{L+2} z^{-L} + h_{2L+2} z^{-2L} + \dots) \quad (1) \\
 &\vdots \\
 &+ z^{-(L-1)} (h_{L-1} + h_{2L-1} z^{-L} + h_{3L-1} z^{-2L} + \dots) \\
 &= \sum_{k=0}^{L-1} z^{-k} H_k(z^L)
 \end{aligned}$$

따라서 위의 (1)식과 같이 분해되어진 필터는 그림 2.의 (a)와 같은 Polyphase 아키텍처를 갖게 된다. 이 아키텍처를 통하여 높은 차수의 필터가 낮은 차수의 독립 필터들로 분산되어 구현될 수 있음을 알 수 있다. 그림 2.의 (b)는 등가 구조를 나타낸다. 이 등가 구조를 이용하여, (a)의 필터와 다운 샘플러는 (c)의 아키텍처로 변형될 수 있다. 이는 병렬로 처리되는 각각의 필터가 지속적으로 처리될 수 있음을 보여준다. 업 샘플러의 경우도 이 장에서 설명한 다운 샘플러의 경우와 같은 방법으로 필터가 Polyphase로 분해되어 아키텍처가 구성되므로 이장에서 설명은 생략하였다.

이와 같은 Polyphase 아키텍처는 디지털 IF와 같은 고속의 필터구현이 요구되어짐에 따라 널리 사용될 수 있는 아키텍처이다. 이는 이 아키텍처가 갖는 병렬처리의 장점과 저속처리의 장점 때문이다. 저속처리의 구조 때문에 반도체로의 구현시 저전력으로 동작 가능하게되어, 휴대용 이동 통신의 핵심 기술로 사용될 수 있을 것이다. 그러나 식 (1)에서 알 수 있듯이, 이 구조는 병렬로 처리되는 각각의 필터 탭수는 원래의 필터 탭수에 비례하게 된다. 즉, 원래의 필터 탭수가 변화하는 경우에, 이 Polyphase 아키텍처의 독립 필터들의 탭수도 비례하여 변화된다. 따라서 본 논문의 III장에서는 병렬

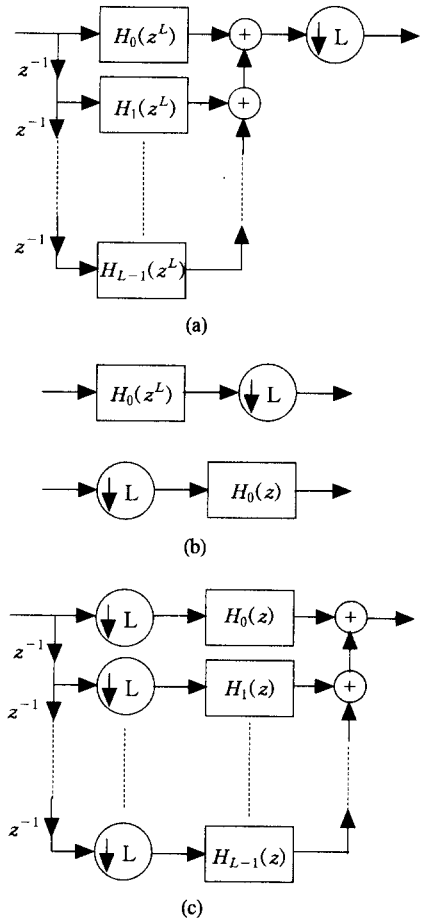


그림 2. (a) 필터의 Polyphase 분해
(b) 등가 구조
(c) 다운 샘플러의 위치이동

처리와 저속처리의 장점을 갖고면서, 처리 단위의 크기 즉 벡터의 크기와 다운 샘플러의 크기가 같은 아키텍처를 제안한다.

III. 제안된 다운 샘플러와 업 샘플러 아키텍처

3.1. 블록 필터링을 이용한 다운 샘플러

원래 블록 필터는 필터의 처리속도를 높이기 위해 만들어진 아키텍처이다.^{[4][5]} 이는 입력신호를 직렬-병렬 변환기를 통하여 벡터 신호로 변환시킨 뒤에 필터링하는 구조로서, 여기에서도 일반적으로 알려져 있는 여러 가지 필터 구조가 역시 존재함이 알려져 있다.^[6] 필터링이 된 후에, 병렬-직렬 변환기를 통하여 스칼라 신호로 바뀌어져야 등가의 필터가 된다. 따라서 그림 1.의 (a)와 같은 다운 샘플러

앞단의 데시메이션 필터는 그림 3.과 같이 블록 필터의 아키텍처로 구현되어질 수 있다.

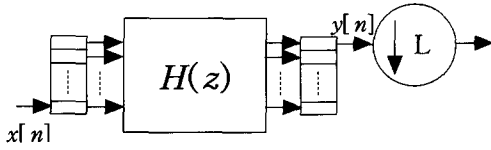


그림 3. 데시메이션 필터의 블록 필터링

그러면 일반적인 FIR 필터의 전달 함수가 어떤 방법으로 블록 필터의 전달 함수로 변형될 수 있는지를 먼저 알아보기로 하자. 블록 필터의 전달 함수는 각 엘리먼트가 다항식으로 표시되는 행렬의 형태를 갖는다. 일반적인 n차의 필터의 전달 함수, z 영역에서의 입출력 관계식, 그리고 시간 영역에서의 입출력 관계식은 다음과 같이 나타내어진다.

$$\begin{aligned}
 H(z) &= h_0 + h_1 z^{-1} + h_2 z^{-2} + \dots + h_N z^{-N}, \\
 Y(z) &= H(z)X(z), \\
 y_n &= h_n * x_n.
 \end{aligned}
 \tag{2}$$

블록 전달함수(Block Transfer Function)를 유도하기 위하여 먼저 위의 시간영역에서의 관계식을 다음의 식 (3)과 같이 행렬식으로 나타내어야 한다.

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_N \\ y_{N+1} \\ y_{N+2} \\ \vdots \end{bmatrix} = \begin{bmatrix} h_0 & 0 & 0 & \dots \\ h_1 & h_0 & 0 & \dots \\ h_2 & h_1 & h_0 & \dots \\ \vdots & \vdots & \vdots & \ddots \\ h_N & h_{N-1} & h_{N-2} & \dots \\ 0 & h_N & h_{N-1} & \dots \\ 0 & 0 & h_N & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ x_{N+1} \\ x_{N+2} \\ \vdots \end{bmatrix}
 \tag{3}$$

첫 번째 단계로서, 위의 행렬식에서 다운 샘플러의 크기 L 과 같은 크기로 중첩되지 않도록 서브 행렬로 분할한다. 즉 벡터는 L 크기의 서브벡터로 행렬은 L x L 크기의 서브행렬로 분할한다. 그러면 다음 식 (4)와 같은 행렬식으로 표현된다.

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_M \\ y_{M+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} H_0 & 0 & \dots \\ H_1 & H_0 & \dots \\ \vdots & \vdots & \ddots \\ H_M & H_{M-1} & \dots \\ 0 & H_M & \dots \\ \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_M \\ x_{M+1} \\ \vdots \end{bmatrix}
 \tag{4}$$

위의 행렬식에서, 벡터들의 엘리먼트들도 모두 서

브 벡터들이며, 행렬의 엘리먼트들도 역시 모두 서브 행렬로 구성되며 다음의 식 (5)와 (6)과 같이 나타내진다.

$$\begin{aligned}
 y_m &= [y_{mL} \ y_{mL+1} \ y_{mL+2} \ \dots \ y_{mL+L-1}]^t \\
 x_m &= [x_{mL} \ x_{mL+1} \ x_{mL+2} \ \dots \ x_{mL+L-1}]^t \\
 m &= 0, 1, 2 \dots
 \end{aligned}
 \tag{5}$$

$$\begin{aligned}
 H_0 &= \begin{bmatrix} h_0 & 0 & \dots & 0 \\ h_1 & h_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{L-1} & h_{L-2} & \dots & h_0 \end{bmatrix}, \\
 H_1 &= \begin{bmatrix} h_L & h_{L-1} & \dots & h_1 \\ h_{L+1} & h_L & \dots & h_2 \\ \vdots & \vdots & \ddots & \vdots \\ h_{2L-1} & h_{2L-2} & \dots & h_L \end{bmatrix}, \dots, \\
 H_M &= \begin{bmatrix} h_N & h_{N-1} & \dots & h_{N-L+1} \\ 0 & h_N & \dots & h_{N-L+2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h_N \end{bmatrix}.
 \end{aligned}
 \tag{6}$$

블록의 크기 L은 다운 샘플러의 크기와 같도록 결정된다. 주어진 원래의 필터 차수 N과 L로부터 블록 필터의 차수 M이 결정되며, 이는 $M = \lceil N/L \rceil$ 로 나타내어진다. 이 괄호는 괄호 안의 값과 같거나 큰, 가장 작은 정수를 의미한다. 이와 같은 블록 행렬식으로부터 다음의 식 (7)과 같은 블록 차등방정식(Block Difference Equation)이 유도되어질 수 있다.

$$y_k = H_0 x_k + H_1 x_{k-1} + \dots + H_M x_{k-M}
 \tag{7}$$

이식에 블록 인덱스 k에 대하여 z-변환을 하면 다음의 블록 전달함수가 얻어진다.

$$H(z) = H_0 + H_1 z^{-1} + \dots + H_M z^{-M}
 \tag{8}$$

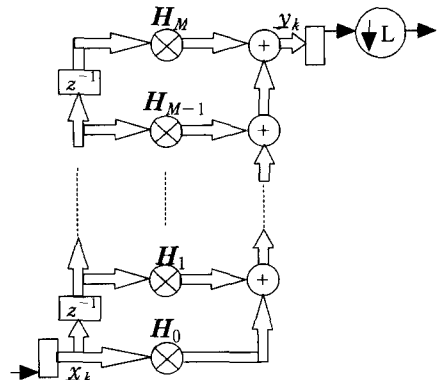


그림 4. 데시메이션의 블록 필터링 아키텍처

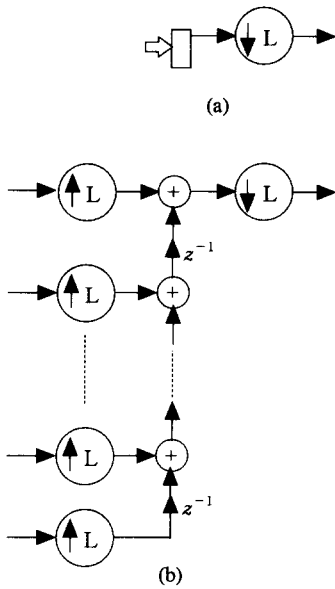


그림 5. (a) 병렬-직렬 변환기와 다운 샘플러 (b) (a)의 등가구조

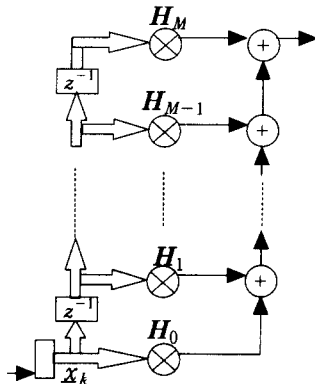


그림 6. 제안된 블록 필터링 다운 샘플러 아키텍처

이렇게 유도된 블록 차등방정식이나 블록 전달함수로부터 다음 그림 4와 같은 Direct Form 구조의 블록 필터 아키텍처가 만들어진다. 그림 4는 입력 단의 직렬-병렬 변환기와 출력 단의 병렬-직렬 변환기, 그리고 다운 샘플러가 포함된 Direct Form 블록 필터링 아키텍처를 보여주고 있다. 이 그림에서 화살표를 굵게 표시한 것은 벡터의 신호이기 때문이다. 일반적인 하나의 입력과 하나의 출력을 갖는 필터의 구성요소와 마찬가지로, 블록 필터의 구조도 곱셈기, 덧셈기, 그리고 지연소자로 구성된다. 이 구

조에서는 곱셈기의 필터계수는 행렬로 표시되어 벡터와 곱해지며, 덧셈기는 벡터끼리의 덧셈이 되며, 지연소자는 벡터의 지연을 의미한다. 특히 곱셈의 양을 살펴보면, 하나의 필터 계수 행렬은 $L \times L$ 로 되어 있으므로 L 크기의 MAC이 L 번 필요하게 된다. 이런 계수가 $M+1$ 개 있으므로 총 L 크기의 MAC이 $L(M+1)$ 번 필요하게 되며, 곱셈의 총 수는 $L^2(M+1)$ 이 된다.

본 논문에서 제안하는 구조 및 계산의 단순화를 이루기 위해서 그림 5와 같은 등가 구조의 제안이 필요하다. 즉 출력 벡터가 병렬-직렬 변환기를 거쳐서 순차신호가 만들어지고 다운 샘플링되는 과정의 (a)는 (b)와 같이 업 샘플러와 지연소자의 조합으로 구성될 수 있다. 그림 5의 (b)에서 알 수 있듯이, L 개의 병렬 입력은 업 샘플러와 지연소자에 의해서 연속되는 순차신호로 만들어진 후에, 다시 다운 샘플링되므로 결국은 L 개의 병렬 신호 중에서 제일 왼쪽의 신호만 남게된다. 따라서 최종적으로 그림 6과 같은 단순화된 블록 필터링 다운 샘플러의 구조가 만들어진다. 여기에서 행렬의 계수와 입력벡터가 곱하여져서 출력 벡터의 맨 왼쪽 행만이 필요하므로 결국은 행렬과 벡터의 곱이 아니라 벡터와 벡터의 곱이 된다. 따라서 각각의 계수 행렬도 맨 위의 행만이 필요하다. 곱셈의 양을 살펴보면, 계수 당 L 크기의 MAC이 1번 요구되며, 계수가 $M+1$ 개 이므로 총 요구되는 곱셈은 $L(M+1)$ 이 된다. 따라서 그림 4의 구조에 비하여 L 분의 1로 곱셈의 양이 감소되어짐을 알 수 있다.

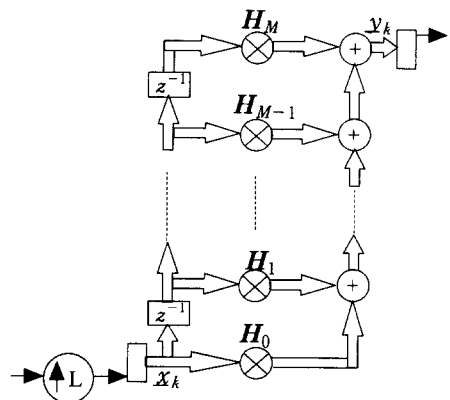


그림 7. 인터플레이션의 블록 필터링 아키텍처

IV장에서 예제를 통하여 비교하겠지만, 그림 6의 구조에서의 곱셈의 양과 II장에서 설명한 기존의

Polyphase 다운 샘플러 아키텍처의 곱셈의 양은 같다. 최종적으로 유도된 그림 6.에서 보여지듯이, 이 블록 필터의 구조는 다운 샘플러와 결합할 때에 매우 유용한 아키텍처가 유도됨을 알 수 있다. 즉 블록 필터의 병렬-직렬 변환기에서 구조적으로 생겨나는 업 샘플러 및 지연소자와 그 뒤에 연결되는 다운 샘플러가 서로 상쇄되어 계산량의 감소로 인한 저속처리가 가능해지며, 입력되는 순차신호의 직렬-병렬 변환기를 통한 병렬처리의 장점도 갖는 구조가 만들어짐을 알 수 있다.

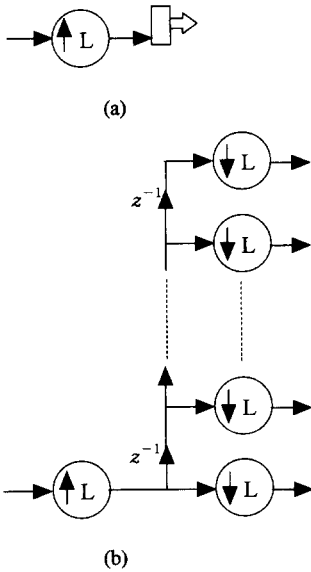


그림 8. (a)업 샘플러와 직렬-병렬 변환기
(b) (a)의 등가구조

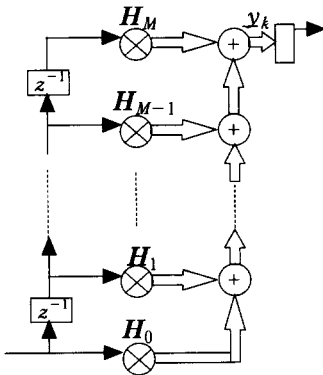


그림 9. 제안된 블록 필터링업 샘플러 아키텍처

3.2. 블록 필터링을 이용한 업 샘플러

이 절에서 제안되는 업 샘플러의 아키텍처도 3.1.

절에서 기술한 다운 샘플러의 아키텍처와 같은 단계를 거쳐서 만들어진다. 그림 7.은 그림 1.의 (b)를 Direct Form의 블록 필터링을 적용하여 만든 구조이다. 여기에서도 다운 샘플러의 경우와 마찬가지로 업 샘플러와 직렬-병렬 변환기의 직렬연결은 그림 8.과 같은 등가구조를 갖는다. 그림 8.의 (b)를 자세히 살펴보면, 업 샘플러를 통하여 L배로 0이 들어가면서 샘플의 수가 늘어난다. 이 순차신호는 지연소자를 거치면서 위로 올라가서 다시 L로 다운 샘플링이 되므로, 결국은 맨 윗쪽의 선으로만 입력신호가 들어가고 나머지 선은 항상 0이 들어가게 된다. 즉 L 크기의 벡터 입력 중에서 첫 번째의 행만 이 값을 갖고 나머지는 모두 0이 되므로 그림 9.와 같이 단순화되어진다.

이 그림에서 행렬로 표현되는 필터계수와 스칼라의 입력의 곱셈은 역시 출력벡터계산을 위해서 L 번의 곱셈이면 완료된다. 즉 행렬의 첫 번째 열과 스칼라 입력과의 곱에 의해서 출력벡터가 얻어진다. 총 M+1개의 필터계수가 사용되므로 필요한 총 곱셈의 수는 L(M+1)이다. 지금까지 이장에서 다운 샘플러와 업 샘플러의 블록 필터링과 연계한 저전력 아키텍처의 유도과정을 제안하였다. 다음 장에서는 예제를 통하여 제안된 구조와 Polyphase 구조를 비교 분석해보기로 한다.

IV. 아키텍처 비교 분석

II장에서 설명한 기존의 Polyphase 구조의 다운 샘플러와 III장에서 본 논문이 제안한 블록 필터링 다운 샘플러를 다음 예제로 비교해 보기로 한다. 예제로서 필터의 차수가 16인 FIR 필터(따라서 필터계수는 17개)를 다운 샘플러의 크기 4로 아키텍처를 만들어 보기로 하자. 이런 FIR 필터와 다운 샘플러를 본 논문이 제안하는 블록 필터링 방법으로 구성하면 그림 10.의 (a)와 같다. 여기에서 맨 앞단의 네모는 물론 직렬-병렬 변환기를 나타낸다. 필터계수 중에서 0으로 된 것들은 삭제해도 되나 modulo한 아키텍처를 나타내기 위해서 그대로 두었다. 이 그림에서 보듯이 다운 샘플러의 크기와 블록의 크기가 같으므로 모든 벡터의 크기는 4가 된다. 여기에서 원래의 필터차수가 증가하더라도 벡터의 크기는 4로서 항상 일정하고 단위블록만 위에 추가하면 된다. 즉 필터 차수가 24가 되면 위에 두 개의 단위 블록을 추가하게 된다. 그림 10.의 (b)는 Polyphase로 구조된 다운 샘플러의 아키텍처이다.

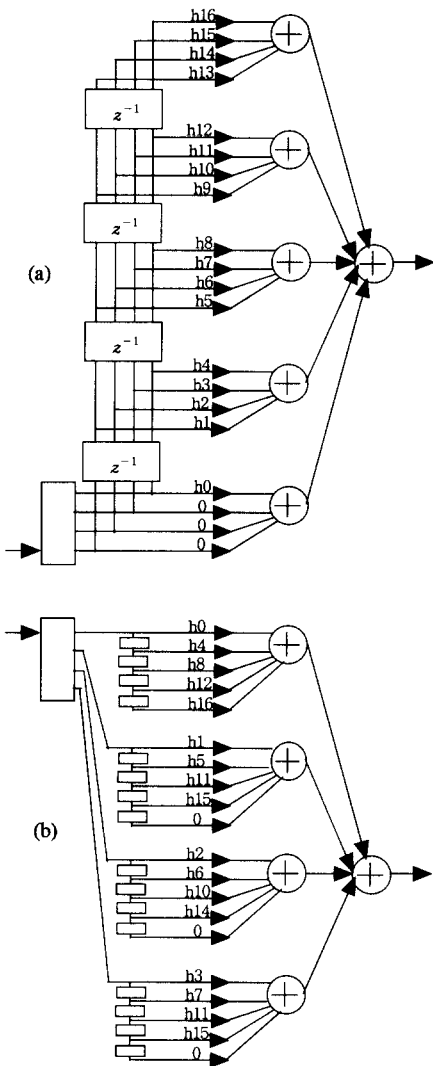


그림 10. 아키텍처의 비교(필터 차수: 16)
 (a) 블록 필터링 다운 샘플러
 (b) Polyphase 다운 샘플러

앞부분의 네모는 역시 직렬-병렬 변환기로서 (a)의 네모와 동일하며, 필터링 안에 포함된 작은 네모는 지연소자를 나타낸다. 이 구조와 블록 필터링의 구조를 비교해보면 많은 유사점이 있음이 보여진다. 즉, 제일 앞단의 직렬-병렬 변환기가 같으며, 뒷단의 2단계의 덧셈기의 구조도 같다. 상이한 점은 Polyphase에서는 병렬화된 각각의 노드가 다시 순차적으로 필터링이 일어나는 구조이다. 이 구조에서는 필터의 차수가 증가하게 되면 각 노드의 서브필터링의 차수가 증가하게 된다.

최근에 상용화되는 벡터 프로세서들은 다수의 입력신호를 벡터화해서 역시 다수의 MAC을 한 사이클에 처리하는 병렬 프로세서이다. 이와 같은 벡터 프로세서로 다운 샘플러를 구현할 때에, 특히 블록 필터링의 구조가 장점을 갖게된다. 그림 10.에서 보여지듯이, 모든 프로세싱의 단위가 벡터로 되며 필터의 차수가 증가하더라도 벡터의 크기는 고정되기 때문이다.

V. 결론

본 논문은 데시메이션 필터가 앞단에 위치한 다운 샘플러 및 인터플레이션 필터가 뒷단에 위치한 업 샘플러의 저전력 블록 필터링 아키텍처를 제안하였다. FIR 블록 필터링 구조가 다운 샘플러와 결합하는 데시메이션의 경우에, 구조적으로 블록 필터링의 뒷단에서 만들어지는 업 샘플러와 따라오는 다운 샘플러가 상쇄되어 저전력의 아키텍처가 만들어짐을 보였다. 물론 업 샘플러와 블록 필터링이 결합하는 인터플레이션의 경우에도 마찬가지로 업 샘플러와 블록 필터링의 앞단에 구조적으로 발생하는 다운 샘플러와 상쇄되어 진다. 이와 같이 하여 Polyphase 아키텍처의 장점인 병렬처리와 저속처리가 모두 가능한 아키텍처가 되며, 샘플러의 크기와 블록의 크기를 일치시킴으로서 modulo한 구조가 됨을 보였다.

참고 문헌

- [1] Y. Neuvo, C. Y. Dong, and S. K. Mitra, "Interpolated finite impulse response filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 563-570, June 1984.
- [2] 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.
- [3] S. J. Jou, S. Y. Wu, and C. K. Wang, "Low-power multirate architecture for IF digital frequency down converter," *IEEE Trans. Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 45, No. 11, pp.

1487-1494, Nov. 1998.

- [4] C. S. Burrus, "Block Implementation of Digital Filters," *IEEE Trans. Circuit Theory*, vol. CT-18, No. 6, pp. 697-701, Nov. 1971.
- [5] 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.
- [6] 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.
- [7] 장영범, *디지털 신호처리 시스템*, 생능출판사, 1998.

장 영 범(Young-Beom Jang)

정회원



1981년 2월 : 연세대학교 전기
공학과 졸업(공학사)
1990년 1월 : Polytechnic
University 전기공학과
졸업(공학석사)
1993년 12월 : Polytechnic
University 전기공학과
졸업(공학박사)

1981년 2월~1987년 12월 : 삼성전자(주) 마이크로
사업부 연구원
1993년 12월~1999년 12월 : 삼성전자(주) System
LSI 사업부 수석연구원
1999년 12월~현재 : 이화여자대학교 정보통신학과
조교수
<주관심 분야> 통신신호처리, 음성/오디오 신호처리

김 낙 명(Nak-Myeong Kim)

정회원



1980년 : 서울대학교 전자공학과
졸업
1982년 : KAIST 전기 및 전자
공학과 석사
1990년 : 미국 Cornell Univer-
sity 전기공학과
공학박사

1990년~1996년 : LG 정보통신(주) 책임연구원
1996년~현재 : 이화여자대학교 공과대학 정보통신학
과 부교수
<주관심 분야> 디지털 이동통신, IMT-2000, 저궤도
이동통신