

---

# Look-up 테이블 기반 펄스성형필터

이창기\* · 임형규\*

## Look-up Table Based Pulse-Shaping Filter

Chang-ki Lee\* · Hyung-kyu Lim\*

### 요약

제안된 구조는 polyphase decomposition과 look-up 테이블 방식을 이용하였다. Synopsys CAD tool을 이용한 모의실험을 통하여 제안한 방식이 기존 방식에 비해 54% 정도의 게이트 수를 감소시킬 수 있고, 9% 정도의 전력소모를 줄일 수 있음을 보인다.

### ABSTRACT

In this paper, an efficient pulse-shaping filter structure for high-density and low-power electronic devices is proposed. The proposed structure is based on polyphase decomposition property and look-up table method. By Synopsys CAD simulations, it is shown that the use of the proposed method can result in reduction in the number of gates by 54% and can reduce power consumption by 9%.

### 키워드

Pulse-shaping Filter, Poly-phase Decomposition, Linear Phase, Look-up Table, Interpolation FIR Filter

## 1. 서론

고집적, 저전력 특성은 모바일 통신 장비와 포터블 멀티미디어 터미널의 중요한 요소들이다. 보간 FIR 필터는 모바일 스테이션에서 요구되어지며, 수신단에서 ISI (Inter-Symbol Interference)를 최소화시키기 위해 기저대역 펄스 성형 필터로 사용된다[1][2]. 펄스 성형 필터는 보간 동작으로 인해 출력 데이터율이 입력 데이터율에 비해 여러배 빠르기 때문에 큰 실리콘 면적과 많은 전력을 소모하게 된다. 따라서, 보간 FIR 필터의 게이트 수와 전력소모를 가능한 최소화시키는 것이 중요하다.

1:4 interpolation 필터의 구현 방식으로는 transversal FIR 필터 디자인 방법과 look-up 테이블 디자인 방법이 사용된다[3][4].

Transversal FIR 필터 방식을 사용할 경우 구현이 용이하다는 장점이 있지만 필터가 요구하는 면적 및 처리 시간이 증가하게 된다. 예를 들어 입력 길이가 1-bit이고 출력 길이가 10-bit인 48탭 transversal FIR 필터를 그림 1과 같이 설계할 경우, 11개의 10-bit 가산기와 44개의 10-bit 레지스터가 필요하게 된다[3]. 그림 1에서 C0, C1, ..., C47은 필터 전달함수  $H(z)$ 의 각 탭 계수를 나타낸다.

Look-up 테이블 방식을 이용한 경우에는 필터 출

---

\* 신경대학교 인터넷정보통신학과  
심사완료일자 : 2009. 04. 21

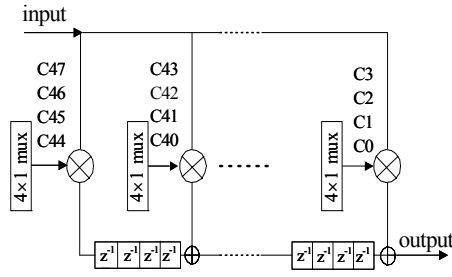


그림 1. Transversal FIR 필터 방식의 구조.  
Fig. 1 Transversal FIR filter structure.

력으로 나올 수 있는 모든 경우 값을 미리 메모리에 저장해 놓았다가 필터의 입력 데이터를 메모리의 주소 입력 선으로 사용하여 출력을 결정하게 된다. 앞에서 설명한 transversal FIR 필터를 이용한 구현 방식을 look-up 테이블 방식으로 변환하기 위해 그림 1의  $H(z)$ 에 polyphase decomposition을 적용하여[5]

$$H(z) = G_0(z^4) + z^{-1}G_1(z^4) + z^{-2}G_2(z^4) + z^{-3}G_3(z^4) \quad (1)$$

으로 분해하면 그림 2와 같은 구조로 구현할 수 있으며 점선 내부를 look-up 테이블로 구성하게 된다. Look-up 테이블은 단지 주소 입력 값에 따라 메모리를 참조하여 출력을 결정해 주기 때문에 그림 1과 같이 많은 연산을 필요로 하는 여러 개의 곱셈기와 덧셈기를 이용한 transversal FIR 필터 방식에 비해 비교적 짧은 시간 내에 필터의 연산 결과를 얻을 수 있다.

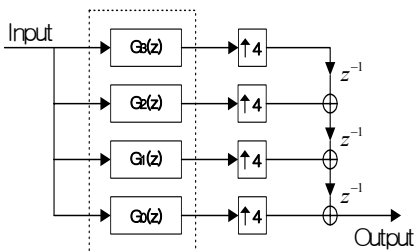


그림 2. Polyphase decomposition을 사용해 재구성된 1:4 interpolation FIR 필터.

Fig. 2 1:4 interpolation FIR filter using polyphase decomposition.

본 논문에서는 polyphase decomposition된 필터의 선형위상 특성을 고려하여 look-up 테이블의 점유면적을 줄일 수 있는 펄스성형필터를 구현한다. II절에서는 look-up 테이블을 이용한 기존의 1:4 interpolation FIR 필터 디자인 방법에 관해 설명한다. III 절에서는 본 논문에서 제안한 필터 설계 방식과 필터를 구성하고 있는 세부 요소들의 구조와 동작 방식에 대해 논의한다. IV절에서는 Synopsys CAD tool을 사용하여 본 논문에서 제안한 방식과 기존의 방식을 이용한 펄스성형필터를 구현하여 칩 점유 면적과 전력 소모에 대한 결과를 비교 검토하며, V 절에서 결론을 맺는다.

## II. Look-up 테이블 필터 설계

$H(z)$ 는 선형위상 특성을 가지므로 식 (1)과 같이 네 개의 서브필터로 분해했을 때 각 서브필터의 계수는 표 1과 같다.

그림 3은 look-up 테이블을 방식을 이용한 1:4 interpolation FIR 필터의 구조를 보여준다[3]. 직렬의 입력 데이터를 병렬로 바꾸어 주어 네 개의 서브 필터링을 동시에 수행하는 look-up 테이블의 입력으로 사용한다.

표 1. 각 서브 필터의 계수  
Table. 1 Coefficient of each sub-filter

$G0$	$C_0 C_4 C_8 C_{12} C_{16} C_{20}$	$C_{23} C_{19} C_{15} C_{11} C_7 C_3$
$G1$	$C_1 C_5 C_9 C_{13} C_{17} C_{21}$	$C_{22} C_{18} C_{14} C_{10} C_6 C_2$
$G2$	$C_2 C_6 C_{10} C_{14} C_{18} C_{22}$	$C_{21} C_{17} C_{13} C_9 C_5 C_1$
$G3$	$C_3 C_7 C_{11} C_{15} C_{19} C_{23}$	$C_{20} C_{16} C_{12} C_8 C_4 C_0$

즉, 그림 1의 transversal FIR 필터 구조와 같이 각 서브 필터의 출력은 열두 개의 계수에 의한 열두 개의 MA(multiplication and accumulation) 연산 결과로 발생된다. 따라서, 열두 개의 데이터가 입력 주소 선으로 사용되어야 한다. 이를 위해 1-bit의 직렬 입력 데이터를 클럭 주기마다 12-bit 쉬프트 레지스터에 저장하여 이를 look-up 테이블의 입력 주소 선으로 사용한다.

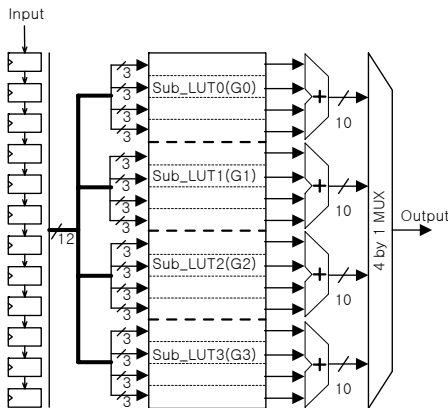


그림 3. Look-up 테이블 방식으로 구성된 1:4 interpolation FIR 필터.  
Fig. 3 1:4 interpolation FIR filter with LUT.

입력 데이터 버퍼의 12-bit 출력은 네 개의 서브 필터링을 수행하는 각 서브 look-up 테이블의 입력으로 사용되므로 필터의 출력이 10-bit 길이를 가질 때, 전체 look-up 테이블은  $4 \times 2^{12} \times 10$  크기를 갖게 된다. 이와 같이 입력 주소 선이 길어지면 메모리의 병목현상 때문에 실제로 구현할 경우에는 입력 주소 선을 그림

3과 같이 분할하여 사용한다. 이 경우 요구되는 전체 look-up 테이블은  $4 \times 4 \times 2^3 \times 10$  크기로 줄어들게 된다. 그러나 각 서브 look-up 테이블의 완성된 출력을 얻기 위해  $4 \times 2$  10-bit carry-save adder와 4개의 10-bit ripple carry adder가 부가적으로 필요하게 된다.

### III. 제안한 Look-up 테이블 설계 방법

본 논문에서는 1:4 interpolation FIR 필터 구현시 면적의 최소화를 위해 표 1의 점선을 기준으로 하여  $G0$  과  $G3$ ,  $G1$ 과  $G2$ 의 계수들이 미리 대칭성을 갖는 특징을 이용한다. 그림 4는 이러한 미리 대칭성을 이용하여 look-up 테이블의 크기를 반으로 줄인 구조를 보여주고 있다<sup>[6]</sup>. 즉, 입력 주소를 그림 3과 같이 12선을 사용하는 대신 6선만을 사용하고  $2 \times 1$  mux를 사용하여 선택신호가 '1'인 경우 첫 번째 쉬프트 레지스터의 출력을 선택하고 '0'인 경우에는 두 번째 쉬프트 레지스터의 출력을 선택한다. 이렇게 선택된 데이터를 look-up 테이블의 입력 주소 값으로 사용하여 기존의 저장 값을 참조하게 된다.

입력되는 주소 선이 많은 경우 look-up 테이블의 병목현상을 피하기 위해 입력 주소 선을 분할하게 되는데 그림 4에서는 주소 선이 여섯 개이므로 세 개씩 분할할 경우 하나의 서브 look-up 테이블은  $2 \times 2^3 \times 10$  크기가 된다. 따라서 그림 3의 구조와 비교하면 look-up 테이블에서 요구되는 점유면적이 반으로 줄게 된다. 그러나 필터의 최종 출력을 얻기 위해 look-up 테이블 다음에 여러 개의 demux, 버퍼, 가산기, mux가 필요하게 된다. 또한 demux와 버퍼를 중심으로 가산기가 분리되어 있으므로 carry-save 방식의 가산기를 사용하기 어려운 단점이 있다. 이러한 점들을 고려하여 제안한 필터 구조는 그림 5와 같다. 계수 대칭성을 이용하여 제안한 그림 5의 각 블록에 대한 구조 및 동작은 다음과 같다.

#### 3.1 입력 레지스터

직렬 입력 데이터를 look-up 테이블의 주소로 사용할 수 있도록 하기 위해 1-bit 입력 스트림을  $clk_1(1.2288MHz)$ 에 맞추어 두 개의 레지스터(그림 5

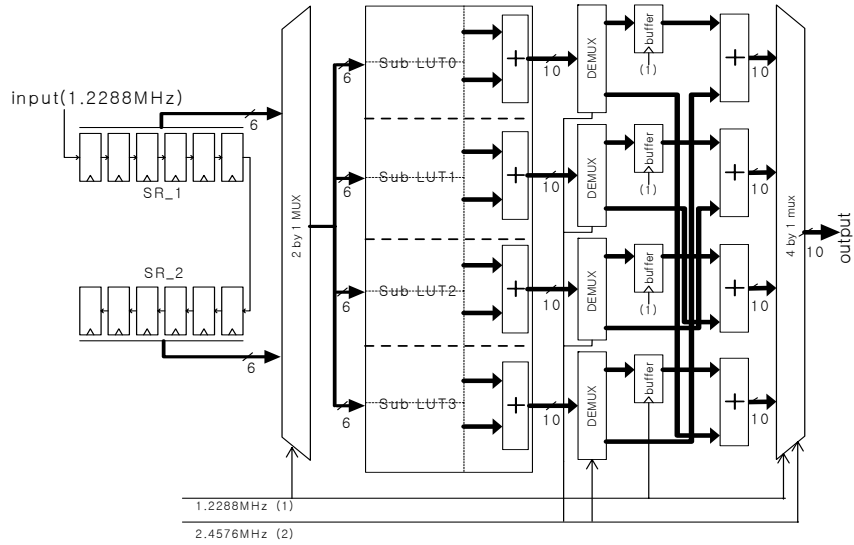


그림 4. 계수대칭성을 이용한 필터구조.  
Fig. 4 LUT structure using symmetric characteristics of coefficients.

의 SR\_1, SR\_2)에 저장시킨다. 각 레지스터는 6-bit 직-병렬 쉬프트 레지스터이다.

### 3.2 2×4 demultiplexer

그림 5의 6-bit 2×4 demux는 clk\_1(1.2288MHz)에 따라 입력 레지스터를 선택하여 6-bit 데이터를 look-up 테이블의 입력으로 전달한다. 즉, demux의 선택신호인 clk\_1이 '1'인 경우 SR\_1의 출력은 demux의 출력단 o1과 o2로 전달되며 SR\_2의 출력은 demux의 출력단 o3과 o4로 전달된다. 또한 clk\_1이 '0'인 경우 SR\_1의 출력은 demux의 출력단 o3과 o4로 전달되며 SR\_2의 출력은 demux의 출력단 o1과 o2로 전달된다. 여기에서 주의해야 할 사항은 표 1에서와 같이 계수들 사이에 미러 대칭 특성을 갖고 있기 때문에 SR\_2의 출력 선을 demux의 입력 선에 반대 순서로 연결시켜 주어야 한다.

### 3.3 서브 필터링을 위한 분할된 look-up 테이블

6-bit의 주소 입력은 그림 5와 같이 2개의 3-bit 주소 입력으로 나뉘어 인가된다. 따라서 각 서브 look-up 테이블은 그림 4와 같이  $2 \times 2^3 \times 10$ 의 크기를 갖게 되어 전체 look-up 테이블의 크기는 그림 3의 구조와 비교하여 반으로 줄어들게 된다. Look-up 테이블은 네 개의 서브 look-up 테이블로 구성된다. 하나의 서브 필터링은 두 개의 서브 look-up 테이블의 출력을 더함으로써 수행된다. 즉, 서브 필터링  $G0$ 과  $G3$ 은 첫 번째와 네 번째 서브 look-up 테이블의 출력 값을 더하여 얻을 수 있고 서브 필터링  $G1$ 과  $G2$ 는 두 번째와 세 번째 서브 look-up 테이블의 출력 값을 더하여 얻을 수 있다. 이를 위해 마지막 단에서 사용된 가산기는 2개의 10-bit carry save adder와 1개의 10-bit ripple-carry adder로 구성된다. 따라서 가산기의 면적도 그림 3의 구조에 비해 1/4로 감소함을 알 수 있다.

본 논문에서는 가산기의 마지막 단을 ripple carry adder로 구현하였지만, 높은 처리속도를 요구하는 시스템에는 가산기의 마지막 단을 carry-look-ahead adder와 같은 고속 가산기로 구현이 가능하다.

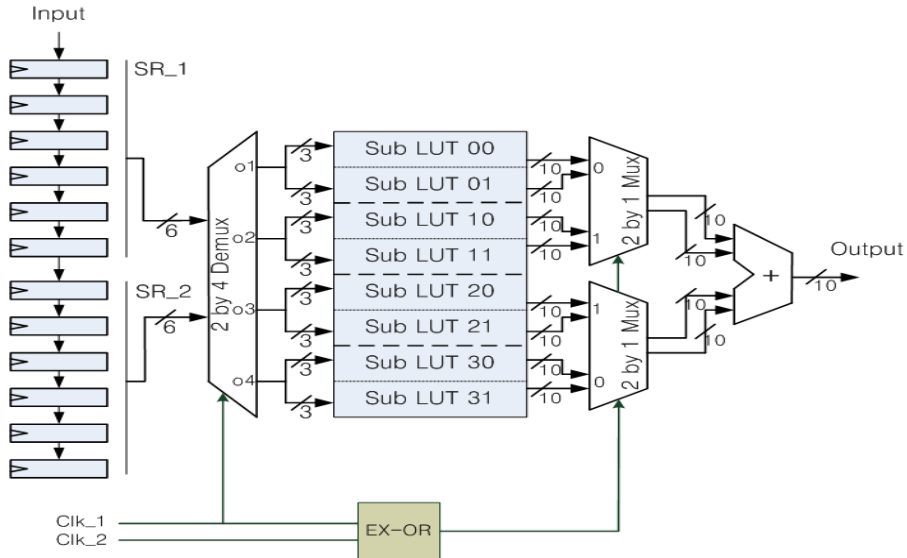


그림 5. 제안한 필터 구조.  
Fig. 5 Proposed LUT filter structure.

### 3.4 2×1 multiplexer

전송을 위해서는 adder\_1과 adder\_2의 가산기에서 출력되는 두 개의 10-bit 신호가 하나의 출력 단을 통해 10-bit 신호로 출력되어야 한다. 이를 수행하기 위해서 2×1 mux를 사용한다. Mux의 선택 신호는 clk\_1(1.2288MHz)과 clk\_2(2.4576MHz)가 사용되었다. clk\_1과 clk\_2가 '00'인 경우 첫째 서브 look-up 테이블 내의 서브 블록을 더해주는 adder\_1(G0)을 선택하여 출력해 준다. '01'인 경우에는 adder\_2(G1), '10'인 경우에는 adder\_2(G2), 그리고 '11'인 경우에는 adder\_1(G3)을 선택하여 출력해 준다.

## IV. 모 의 실 험

그림 1의 transversal 필터 방식과 그림 3과 같은 기존의 look-up 테이블 방식, 그리고 그림 5의 제안된 look-up 테이블 방식을 이용하여 세 종류의 1:4 interpolation 필터를 설계하였다. 필터는 VHDL을 이용하여 동작을 기술하였으며 Synopsys CAD tool을 사용하여 회로를 합성하고 게이트 수준에서 소모되

는 전력을 계산하였다. 세 가지 구현 방식에 대한 면적 요구량과 전력 소모량은 표 2와 같다.

면적을 비교해 보면 transversal 필터 방식은 6774개의 게이트로써 구현되어지며, 기존의 look-up 테이블 방식을 이용한 경우에는 2382개의 게이트가 요구되어지고 본 논문에서 제안한 방식을 이용한 경우에는 1096개의 게이트가 요구된다. 따라서 look-up 테이블 방식을 이용할 경우 transversal 필터 방식보다 적은 게이트를 사용하여 원하는 필터 동작을 수행할 수 있음을 알 수 있다. 또한 제안한 구조에서는 기존의 look-up 테이블 방식에 비해 look-up 테이블 자체만으로도 50%의 면적 감소 효과를 얻을 수 있다. 또한 마지막 단에서 사용된 가산기는 기존의 구조에 비해 1/4로 면적이 감소되었다.

전력 소모를 살펴보면 면적 비교의 경우와 같이 transversal 필터 방식이 look-up 테이블 방식을 이용한 경우에 비해 약 3배 정도의 전력을 사용하는 것을 확인할 수 있다. 제안한 방식으로 구현된 필터가 기존의 look-up 테이블 방식에 비해 9% 정도만의 전력을 감소시켰는데, 이는 제안한 구조의 look-up 테이블의 내부동작속도가 원래 클럭 속도의 2배여서 면적 감소만큼의 큰 효과를 얻지는 못했다.

표 2. 면적과 전력소모 비교

Table. 2 Comparison of area and power consumption

	Transversal 필터 설계 방식	기존 Look-up 테이블 설계 방식	제안한 Look-up 테이블 설계 방식
Area(gate수)	6774	2382	1096
Power(uW)	31.6813	10.2083	9.2896

저자 소개

**이창기(Chang-ki Lee)**

한국전자통신학회 논문지 제 3권 3호 참조

**임형규(Hyung-kyu Lim)**

한국전자통신학회 논문지 제 3권 3호 참조

**V. 결 론**

통신 시스템의 송신 단에 들어가는 펄스 성형 필터의 집적화 및 저전력화를 위한 새로운 구현 방식을 제안하였다. 제안한 방식에서는 decomposition된 필터 계수의 선형위상 특징에 따른 계수 대칭성을 이용하여 look-up 테이블이 차지하는 면적을 반으로 감소시켰다. CAD tool의 모의실험 결과 제안한 구조가 기존의 look-up 테이블 구조에 비해 칩 면적을 54% 정도 감소시키고, 9% 정도의 전력 소모를 감소시킬 수 있었다.

**참고 문헌**

- [1] Lee, Miller, CDMA Systems Engineering Handbook, Artech House Publishers, 1998.
- [2] Bernard Sklar, Digital Communication Fundamentals and Applications, Prentice Hall, 1998.
- [3] In Kang, et al., "Multiple 1:N interpolation FIR filter Design Based on a single architecture," ISCAS 98, Monterey(CA).
- [4] J. Proakis and D. Manolakis, Digital Signal Processing: Principles, Algorithms, and Applications, Prentice Hall, 1996.
- [5] P.P Vaidyanathan, Multirate Systems and Filter Banks, Prentice Hall, 1993.
- [6] 김명순, 김대익, 정진균, 임명섭, "SSB/BPSK-DS/CDMA를 위한 1:N Interpolation FIR 필터 디자인," JCCI 2000, 2000. 5.