

# 곱셈기가 필요없는 2의 누승 계수를 사용한 프로그램 가능한 FIR필터의 구현

正會員 吳 宇 鎭\* 正會員 李 勇 勳\*

## Implementation of Programmable Multiplierless FIR Filters with Powers-of-Two Coefficients

Woo Jin Oh\*, Yong Hoon Lee\*, *Regular Members*

### 요 약

본 논문에서는 계수가 2의 누승으로 표현되는 프로그램 가능한 FIR 필터 (2PFIR 필터)를 구현할때 유용한 성질을 구했다. 특히 2PFIR 필터의 계수가 갖는 2의 누승 값이 M ternary의 canonical signed digit (CSD) 코드로 표현되면 ternary 값을 M개보다 적은  $\{0, 1, 2, \dots, M-1\}$ 의 부분집합에서 선택해도 같은 결과가 얻어진다는 사실을 보였다. 따라서 프로그램 가능한 2PFIR 필터의 쉬프터 길이가 M보다 작아지므로 훨씬 효율적으로 구현할 수 있다. 또한 본 논문에서는 몇 가지 실험 결과를 통해 2PFIR 필터의 쉬프터 길이를 추가적으로 감축할 수 있음을 보였다.

### ABSTRACT

An observation which is useful for hardware implementation of programmable FIR filters with powers-of-two coefficients (2PFIR filters) is made. Specifically, it is shown that the exponents of filter coefficients representable by the canonical signed digit(CSD) code with M ternary digits can be chosen from some subsets of  $\{0, 1, \dots, M-1\}$ . This observation naturally leads to 2PFIR filters with shorter shifters whose length is strictly less than M and, as a consequence, leads to an efficient hardware structure for programmable 2PFIR filtering. In addition, we present some experimental results indicating that the shifters of 2PFIR filters can be shortened further in some cases.

\* 韓國科學技術院 電氣, 電子工學科  
Dept. of Electrical and Electronics Eng., KIST  
論文番號 : 94175  
接受日字 : 1994年 7月 2日

## I. 서 론

2의 누승 계수를 갖는 FIR 필터, 즉 2PFIR 필터는 구현이 간단하여 신호 처리 분야에서 많은 관심을 받아 왔다.<sup>[1][11]</sup> 2PFIR 필터는 계수가 2의 누승의 합과 차로 표현되므로 각 탭의 곱셈이 쉬프트(shift)와 덧셈만으로 간단히 구현된다.

특히, 이러한 2PFIR 필터는 각 계수가 정해진 용도에만 사용되도록 고정되면 쉬프트가 단순한 연결인 hard-wired 쉬프트로 구현되어 더욱 간단해진다. 반면에 프로그램 가능한 2PFIR 필터를 구현할 때는<sup>[10][11]</sup> 배럴 쉬프트(barrel shifter), 쉬프트 레지스터(shift register) 또는 preshifter<sup>[11]</sup>와 같이 쉬프트 양을 가변할 수 있는 쉬프트가 필요하여 고정 계수보다 훨씬 복잡해질 뿐만 아니라 수행 속도가 저하 될 수도 있다.

본 논문에서는 2PFIR 필터의 계수가 갖는 2의 누승 값이 M ternary의 canonical signed digit(CSD) 코드로 표현되는 사실로부터 ternary 값을 M개의 {0, 1, 2, ..., M-1} 집합보다 적은 그의 부분집합에서 선택해도 같은 결과가 얻어진다는 성질을 찾았다. 이 성질은 프로그램 가능한 2PFIR 필터를 구현하는데 필요한 쉬프트의 길이가 짧아지는 것을 뜻하며, 따라서 직접 구현하는 것보다 효율적으로 구현된다는 것을 의미한다.

또한 2PFIR 필터의 각 탭을 구현하는데 필요한 쉬프트의 갯수와 길이에 따른 상관 관계를 이용하여 쉬프트의 길이를 추가적으로 감축할 수 있는 방법을 본 논문에서 제안 하였다. 즉 짧은 길이의 쉬프트를 사용하는 대신에 쉬프트의 갯수를 증가시켜 전체적인 복잡도는 줄이면서 성능을 개선하는 방법과 약간의 성능 저하를 감수하면서 훨씬 작은 쉬프트를 사용하는 두가지 방법을 고려하였다. 몇개의 예제를 통하여 쉬프트의 갯수와 길이를 변화하며 여러가지 필터의 성능을 비교 하겠다.

본 논문의 구성은 다음과 같다. 2장에서는 CSD 코드의 특징을 살펴보고 위의 성질을 이용하여 프로그램 가능한 2PFIR 필터의 쉬프트 길이를 줄일 수 있음을 설명하겠다. 3장에서는 2PFIR 필터의 쉬프트를 훨씬 더 줄였을 때의 몇 가지 실험 결과를 통해 구현이 간단하면서도 더 좋은 성능을 갖는 것을 보이겠다.

## II. 프로그램 가능한 2PFIR 필터의 효율적 구현

2PFIR 필터의 임펄스 응답,  $h(m)$ 은 2의 누승의 합과 차로 다음과 같이 나타낸다.

$$h(m) = \sum_{k=1}^L S_k 2^{-P_k} \quad (1)$$

여기서  $S_k \in \{-1, 0, 1\}$ ,  $P_k \in \{0, 1, \dots, M-1\}$  이고, M은 ternary 값이 가질 수 있는 자릿수이고 L은 영이 아닌 자릿수의 갯수이다. 이 표현은 radix-2 signed digit 코드로 알려져 있으며 일반적으로 주어진  $h(m)$ 에 대하여 여러 가지의 signed digit 표현이 존재한다.<sup>[12][13]</sup>

이 signed-digit 코드를 주어진 값에 대해서 유일하게 표현하는 방법은 아래에서 설명할 CSD코드이다. 이 CSD코드는 0이 아닌 자릿수의 갯수인 L을 최소화하며 부가적으로 (1)식의 영이 아닌 자리의 위치가 인접하지 않아야 한다. 즉, 어떤 i와 j에 대해서 i+j인 경우에 다음의 식이 성립해야 한다.<sup>[12]</sup>

$$|P_i - P_j| \geq 2 \quad (2)$$

예를 들면, 0.375의 CSD 표현은  $2^{-1}-2^{-2}+2^{-3}$ 과  $2^{-2}+2^{-3}$ 이 아니라  $2^{-1}-2^{-3}$ 뿐이다.

식 (1)의 2PFIR필터 계수,  $h(m)$ 과 입력,  $x(n-m)$ 의 곱은 그림 1과 같이 곱셈기가 필요 없이 덧셈기와 쉬프트로 구현이 가능하다. 그림 1은 프로그램 가능한 2PFIR필터를 구현하는 전체적인 구조이며, 그림 1(b)와 (c)는 각 탭을 구현하는 두 가지 예를 보였다. 그림 1(b)는 각각의 곱  $x(n-m)2^{-P_k}$ 를 배럴 쉬프트 또는 쉬프트 레지스터로 구현한 것으로서, 이 쉬프트들은 식 (1)의 각  $P_k$ 가 일반적으로  $\{0, 1, \dots, M-1\}$ 의 집합에서 선택되므로 N 비트의 입력에 대해  $N \times M$  비트가 필요하다. 그림 1(c)의 구조는 preshifter<sup>[11]</sup>라 불리는 M비트의 hard wired 쉬프트에 프로그램 가능한 multiplexer로  $\{x(n-m), x(n-m)2^1, x(n-m)2^2, \dots, x(n-m)2^{(M-1)}\}$ 에서 k에 따라  $x(n-m)2^{-P_k}$ 를 선택하게 되어 있다. 다음에서는 CSD의 성질을 응용하여 그림 1(b)와 (c)에서 사용된 쉬프트의 길이를 줄이는 방법에 대해 설명 하겠다.

임의의 M과 L에 대해서 (1)식에서 얻어지는  $-1$ 과 1 사이의 모든 수들의 집합을  $S_{M,L}$ 이라 하자. 따라서 이 집합,  $S_{M,L}$ 의 원소는  $[-1, 1]$ 영역 내에서 CSD로 표현할 수 있는 모든 수이며, 일 예로  $S_{2,1}$ 는  $\{-1, -0.5, 0, 0.5, 1\}$ 이 된다.  $S_{M,L}$ 은 (1)식의  $P_k$ 를  $\{0, 1, \dots, M-1\}$ 의 집합에서 선택하여 얻어진 것이나, 다음에서  $P_k$ 를  $\{0, 1, 2, \dots, M-1\}$ 의 적절한 부분 집합에서 선택해도 동일한  $S_{M,L}$ 을 얻을 수 있음을 보이겠다.

성질 :  $Z_{M,L}(k) \subset \{0, 1, \dots, M-1\}$ ,  $1 \leq k \leq L$ 을 아래의 (3)식으로 주어진 연속적인 정수의 집합이라 하면,

$$Z_{M,L}(k) = \{2(k-1), 2(k-1)+1, \dots, (M-1)-2(L-k)\} \quad (3)$$

$S_{M,L}$ 은  $P_k \in Z_{M,L}(k)$ 일 때 들어진다.

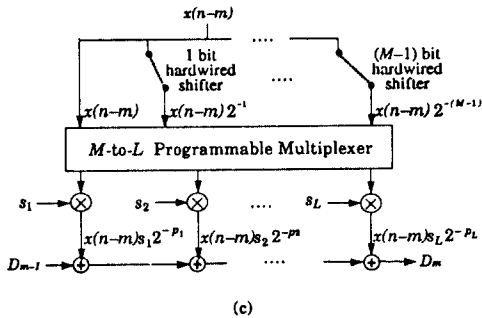
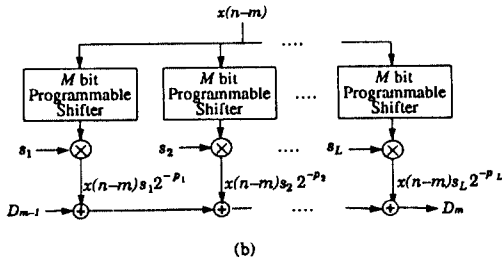
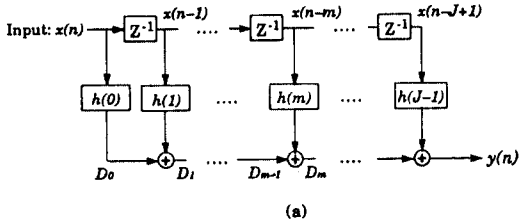


그림 1 (a) J개의 탭을 가진 2PFIR 필터를 구현한 예, (b)와 (c)  $D_m = D_{m-1} + h(m) \times (n-m)$ 을 계산하는 각 탭을 구현하는 두가지 구조

증명 : 일반적으로  $P_1 < P_2 < \dots < P_L$ 로 표현할 수 있으므로 0과 1을  $Z_{M,L}(1)$ 의 원소라고 가정하자.  $|P_1 - P_2| \geq 2$ ,  $P_1 < P_2 < \dots < P_L$  과  $0 \in Z_{M,L}(1)$ 이므로  $k \geq 2$ 인 모든  $P_k$ 는 2이상이어야 한다. 따라서 모든  $Z_{M,L}(k)$ ,  $k \geq 2$ 는 0과 1을 포함할 필요가 없어진다. 또한 2가  $Z_{M,L}(2)$ 에 포함되면  $Z_{M,L}(k)$ ,  $k \geq 3$ 에서 3과 4를 뺄 수 있다. 같

은 방법으로  $\{0, 1, \dots, 2(k-1)-1\}$ 을  $Z_{M,L}(k)$ ,  $k \geq 2$ 에서 제외시킬 수 있다. 여기서  $Z_{M,L}(L) = \{2(L-1), 2(L-1)+1, \dots, M-1\}$ 을 생각해보자.  $P_1 < P_2 < \dots < P_L$ 이고  $(M-1) \in Z_{M,L}(L)$ 이므로  $|P_1 - P_L| \geq 2$ 의 조건을 이용하여  $(M-1)$ 과  $(M-2)$ 를  $Z_{M,L}(k)$ ,  $k \leq L-1$ 에서 제거할 수 있다. 같은 방법으로  $\{(M-1)-2(L-k)+1, \dots, M-1\}$ 들의 원소는 집합  $Z_{M,L}(k)$ ,  $1 \leq k \leq L-1$ 의 원소가 될 수 없다.

이 성질로부터  $Z_{M,L}(k)$ 의 집합이 갖는 원소 갯수는  $M-2L+2$ 이 되므로 그림 1(b)의  $N \times M$  비트 쉬프터는  $(M-2L+2)$ 비트로 줄어들고 그림 1(c)의 hard-wired 쉬프터는  $(2L-1)$ 비트 만큼 간단해진다. 그림 2(a)와 (b)는 각각 그림 1(b)와 (c)에 이 성질을 적용하여 개선된 형태이다. 그림 2(a)는  $x(n-m)$ 과  $2^{-P_i}$ 의 곱을  $2(k-1)$  비트 hard-wired 쉬프터를 거친 후에  $(M-2L+2)$ 비트 쉬프터를 통과하게 된다. 여기서 각각의 프로그램 가능한 쉬프터 길이는  $2(L-1)$ 만큼씩 줄어드는 장점이 있다. 쉬프터의 길이가 짧아짐에 따라 쉬프트 레지스터를 사용할 때는 처리 속도가 비례적으로 증가하고, 배럴 쉬프터를 사용할 경우에는 구현의 복잡도가 줄어들게 된다. 예를 들면,  $(M, L) = (12, 2)$ 과  $(12, 3)$ 일때 각각  $2(L-1) = 2$ 와 4만큼 줄어든다. 따라서  $(12, 3)$ 의 경우 쉬프트 레지스터를 사용한 경우 처리 속도가  $2(L-1)/M = 1/3$ , 즉 33%의 처리 속도가 증가한다. 그림 2(b)에서는  $\{x(n-m) \mid 0 \leq i \leq M-2L+1\}$ 가  $M-2L+2$  비트 hard-wired 쉬프터로 구현하고  $x(n-m)2^{-P_i}$ 를 각  $k$ 에 대해서 프로그램 가능한 multiplexer로 선택하여  $2(k-1)$  비트 hard-wired 쉬프터를 거친 후  $x(n-m)2^{-P_i}$ 가 얻어진다. 따라서 이 구조는 그림 1(c)보다 간단한 구조를 갖고 있다.

### III. 쉬프터 길이의 추가적 감축

앞 절에서는 쉬프터의 길이를 2PFIR 필터의 성능 저하 없이 CSD의 성질을 이용하여 줄이는 방법에 대해서 설명했다. 본 절에서는 각 탭을 구현하기 위해 필요한 쉬프터의 갯수와 길이에 따른 상호 관계를 이용하여 전체 구현을 더욱 간단히 하는 방법에 대해서 설명하겠다. 쉬프터의 길이,  $P_k$ 를  $M-2L+2$ 보다 더 줄이는 방법은 각  $P_k$ 를  $Z_{M,L}(k)$ 의 부분 집합, 즉  $P$ 개의 연속적인 정수로 구성된  $Z'_{M,L}(k)$ 에서 선택하면 된다.  $Z'_{M,L}(1)$ 과  $Z'_{M,L}(L)$ 을 각각  $\{0, 1, \dots, P-1\}$ 과  $\{M-P, \dots, M-1\}$ 으로 놓자.  $P_k \in Z'_{M,L}(k)$ 일때 (1)식으로 표현 가능한 모든

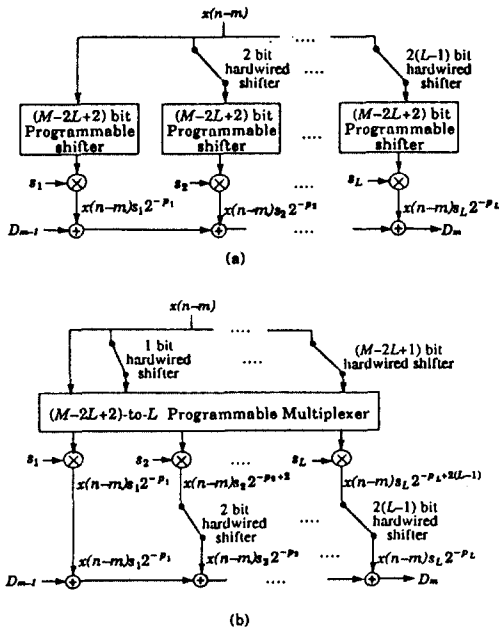


그림 2. 그림 1(b)와 (c)의 구조를 개선한 형태

수를  $S'_{ML}$ 라 하면 당연히  $S'_{ML}$ 은  $S_{ML}$ 의 부분 집합이 된다. 몇 개의 예제를 통해서 어떤 경우에는  $S'_{ML}$ 에서 설계된 2PFIR 필터가  $S_{M,0L}$ ,  $i \geq 1$ 으로 설계된 경우보다 훨씬 간단한 구조를 가지면서도 더 좋은 성능을 갖고 있다.

표 1.  $S_{ML}$ 과  $S'_{ML}$ 의 원소 수 및 여러가지 2PHR 필터에 따른 최대 ripple 값. (a) L=3, (b) L=2,

			$S_{10,3}$	$S_{12,3}$	$S'_{12,3}$
		$S_{ML}$ 과 $S'_{ML}$ 의 원소 수	513	1041	777
최대 ripple (dB)	LPF	J=31	48.80	52.02	49.16
		J=37	49.16	54.75	50.40
	HPF	J=31	47.84	51.66	50.40
		J=37	48.43	57.04	52.21
	BPF	J=31	47.79	52.47	52.32
		J=37	48.83	58.96	55.36
	BSF	J=31	49.17	52.01	49.38
		J=37	48.63	58.89	52.14

(a)

			$S_{10,3}$	$S_{12,3}$	$S'_{16,2}$	$S'_{12,3}$
		$S_{ML}$ 과 $S'_{ML}$ 의 원소 수	149	225	425	205
최대 ripple (dB)	LPF	J=31	41.59	42.36	42.36	39.46
		J=37	41.65	43.80	43.80	43.10
	HPF	J=31	43.63	45.55	47.13	45.37
		J=37	44.20	46.24	47.56	43.97
	BPF	J=31	45.76	46.48	47.85	44.35
		J=37	46.38	48.68	48.68	50.35
	BSF	J=31	45.09	48.29	49.31	45.08
		J=37	45.09	50.31	50.31	48.09

(b)

본 논문에서 설계된 4가지의 2PFIR 필터는 [2]에서 사용한 것으로서, 저역 통과 필터( $f_0=0.15$ ,  $f_s=0.25$ ), 고역 통과 필터( $f_0=[0.25, 0.35]$ ,  $f_s=[0.15, 0.45]$ ) 등이다. 여기서  $f_0$ 와  $f_s$ 는 각각 정규화된 통과 대역 주파수와 저지 대역 주파수이고, 각각의 대역에서 ripple 가중치는 1.0이다. 2PFIR 필터를 [2]에서 제시한 국부 탐색(local search)방법으로 여러 가지 (M, L)에 따라서 31 탭과 37 탭으로 설계했다. 표 1에 설계된 2PFIR 필터의 최대 ripple을 보였으며, 이때의  $S'_{ML}$ 에 대한  $Z'_{ML}(k)$ 은 각각 아래와 같다.

(M, L)=(12, 3)일 때,

$$\begin{aligned}
 Z_{12,3}(1) &= \{0, 1, \dots, 4\} \\
 Z_{12,3}(2) &= \{4, 5, \dots, 8\} \\
 Z_{12,3}(3) &= \{7, 8, \dots, 11\}
 \end{aligned} \tag{4}$$

이고 (M, L)=(12, 2)일 때,

$$\begin{aligned}
 Z_{12,2}(1) &= \{0, 1, \dots, 7\} \\
 Z_{12,2}(2) &= \{4, 5, \dots, 11\}
 \end{aligned} \tag{5}$$

이 된다. 표 1(a)는 L=3일 때로서,  $S'_{12,3}$ 은  $S_{10,3}$ 보다 더 많은 원소를 갖고 있을 뿐만 아니라  $S'_{12,3}$ 에서 설계된 필터가  $S_{10,3}$ 의 것 보다 더 좋은 성능을 보인다.  $Z'_{12,3}$ 은

5개의 원소를  $Z_{10,3}$ 은 6개의 원소를 갖고 있으므로  $S'_{12,3}$ 에서 설계된 필터가  $S_{10,3}$ 의 것 보다 더 짧은 쉬프트를 필요로 한다. 따라서 대부분의 경우에  $S'_{12,3}$ 이  $S_{10,3}$ 보다 더 선호 될 것이다.

표 1(b)는  $L=2$ 인 경우로  $S'_{12,2}$ 는  $S_{10,2}$ 보다 더 많은 원소를 갖고 있지만  $S'_{12,2}$ 에서 설계된 필터는  $S_{10,2}$ 보다 더 큰 ripple을 보이는 경우가 있다. 그러므로  $S'_{12,2}$ 는  $S_{10,1}$ 보다 항상 선호될 수 없으며  $L=2$ 인 경우에는 쉬프트의 길이를 추가로 줄이는 이 방법이 권장할 만한 방법이 아닌 것으로 믿어진다.

마지막으로,  $S'_{12,3}$ 와  $S_{16,2}$ 의 관계를 살펴보자.  $S'_{12,3}$ 은  $S_{16,2}$ 보다 짧은 쉬프트 길이를 필요로 하고 있으나 각각의 집합에서 설계된 필터는  $S'_{12,3}$ 의 것이 훨씬 더 좋을 것을 표 1에서 알 수 있다. 따라서  $S'_{12,3}$ 은  $S_{16,2}$ 를 대체하는 효율적인 집합으로 보인다. 다만 대부분의 경우에 쉬프트의 길이가 짧아지면 처리속도가 빨라지나,  $L$ 이 커지면 구현의 복잡도(칩 면적, 게이트 수 등)가 증가하므로 필요에 따라  $S'_{12,3}$ 와  $S_{16,2}$ 에서 적절히 선택해야 한다.

#### IV. 결 론

본 논문에서는  $L$ 개의 영이 아닌 자릿수를 갖는(CSD) 코드를  $M$  ternary 값으로 표현할 때 이 ternary 값을  $\{0, 1, \dots, M-1\}$ 의 부분 집합인  $M-2L+2$ 개의 연속적인 정수 집합에서 선택해도 본래의 집합에서 얻은 것과 같다는 성질을 찾아냈다. 이 성질로부터 계수가 CSD값으로 표현되는 프로그램 가능한 2PFIR 필터를 성능의 저하 없이 쉬프트의 길이를  $M$ 에서  $2(L-1)$ 만큼 썩 줄일 수 있음을 보였다. 또한 몇 개의 예제를 통해서 쉬프트의 길이를 추가적으로 더 줄이는 몇 가지 방법을 보였다.

#### 참 고 문 헌

[1] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 583-591, June 1983.

[2] Q. Zhao and Y. Tadokoro, "A simple design of FIR with powers-of two coefficients," *IEEE Trans. Circuits Syst.*, vol. CAS-35, pp. 566-570, May 1988.

[3] H. Samuelli, "An improved search algorithm for

the design of multiplierless FIR filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. CAS-36, pp. 1044-1047, July 1989.

[4] Y. C. Lim, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," *IEEE Trans. Circuits Syst.*, vol. CAS-37, pp. 1480-1486, Dec 1990.

[5] N. Benvenuto, M. Marchesi and A. Uncini, "Application of simulated annealing for the design of special digital filters," *IEEE Trans. Signal Processing*, vol. SP-40, pp. 323-332, Feb 1992.

[6] M. Ishikawa et al., "Automatic layout synthesis for FIR filters using a silicon compiler," in *Proc. 1990 IEEE Int. Symp. Circuits Syst.*, pp. 2588-2591, May 1990, New Orleans USA.

[7] T. Yoshino, R. Jain, P. T. Yang, H. Davis, W. Gass and A. H. Shah, "A 100-MHz 64-tap FIR digital filter in 0.8- $\mu$ m BiCMOS gate array," *IEEE J. Solid-State Circuits*, vol. 25, pp. 1494-1501, Dec 1990.

[8] R. Jain, P. T. Yang and T. Yoshino, "FIRGEN: A computer-aided design system for high performance FIR filter integrated circuits," *IEEE Trans. Signal Processing*, vol. 39, pp. 1655-1668, July 1991.

[9] J. B. Evans, "An efficient FIR filter architecture," in *Proc. 1993 IEEE Int. Symp. Circuits Syst.*, pp. 627-630, May 1993, Chicago USA.

[10] J. B. Evans, Y. C. Lim and B. Liu, "A high speed programmable digital FIR filter," in *Proc. 1990 IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 969-971, Apr 1990, Albuquerque USA.

[11] K. Khoo, A. Kwentus and A. N. Willson, Jr., "An efficient 175MHz programmable FIR digital filter," in *Proc. 1993 IEEE Int. Symp. Circuits Syst.*, pp. 72-75, May 1993, Chicago USA.

[12] F. J. Taylor, *Digital Filter Design Handbook*. New York: Marcel Dekker, Inc., 1983.

[13] K. Hwang, *Computer Arithmetic: Principles, Architecture, and Design*. New York: Wiley, 1979.

吳宇鎭 (Woo Jin Oh) 정회원

1966년 3월 2일생

1989년 2월 : 한양대학교 전자공학 과(학사)

1991년 8월 : 한국과학기술원 전자공학과 (석사)

1991년 9월 ~ 현재 : 한국과학기술원 박사과정 재학중

※주관심분야 : 선형 필터, VLSI신호처리

李勇勳 (Yong Hoon Lee) 정회원

1955년 7월 12일생

1978년 2월 : 서울대학교 전기공학 과(학사)

1980년 2월 : 서울대학교 전기공학 과 대학원(석사)

1984년 8월 : Univ. of Pennsylvania (박사)

1984년 ~ 1989년 SUNY at Buffalo 조교수

1989년 ~ 현재 : 한국과학기술원 부교수로 재직

※주관심분야 : VLSI신호처리, 선형 및 비선형 필터, 통신시스템