

무선 이동통신을 위한 효율적인 TDL 필터 설계 및 성능 분석

정회원 서승완*, 선우 명훈*

Design of an Efficient TDL Filter for Wireless Communications and its Performance Evaluation

Seung Wan Seo*, Myung Hoon Sunwoo* *Regular Members*

요 약

기존 direct-form 필터는 탭 수가 증가하면 한 사이클에 여러 비트의 연산을 수행하므로 저속으로 동작한다. 동작속도를 높이기 위해 파이프라인 방식을 적용하는 transpose-form 필터를 사용한다. 그러나 transpose-form 필터는 많은 게이트를 사용한다. 본 논문은 시간지연과 하드웨어 면적을 고려한 설계를 이용하여 개선된 tapped delay line (TDL) 필터를 제안하며 성능 분석을 수행한다. 제안된 필터는 효율적인 파이프라인을 적용하여 direct-form 필터에 비하여 하드웨어 크기는 증가하나 고속의 연산이 가능하며 transpose-form 필터에 비하여 연산속도는 같으며 하드웨어가 적게드는 장점을 가진다. 제안된 TDL 필터 설계 방식을 디지털 이동통신의 표준안인 IS-95에 적용하여 128 탭 정합 필터에 사용할 경우, 연산시간을 transpose-form 필터에 비하여 30% 이상 게이트가 줄어들고 임계경로가 같으므로 연산속도는 같다. 또한 direct-form 필터에 비하여 게이트 수는 88% 증가하나 연산 속도는 2 배 이상 증가한다.

ABSTRACT

Since the direct-form filter sums several bits at one cycle, it should operate at low speed when the number of taps is increased. To increase the operating frequency, the transpose-form filter using a pipeline scheme has been proposed. However, the transpose-form filter requires a large number of gates. This paper proposes a tapped delay line (TDL) filter considering time delay and hardware size and evaluates performance. Although the proposed filter using efficient pipeline schemes requires larger number of gates compared with direct-form filter, it increase the clock speed compared with direct-form filter and requires less number of gate compared with transpose-form filter. Applying the proposed TDL filter of 128 taps for the CDMA standard, IS-95, can reduce the gate count by more than 30% compared with the transpose-form filter without sacrificing the clock speed. In addition, although the proposed TDL filter increases the gate count about 88% with the direct-form filter, that can increase the clock speed about 200% compared with the direct-form filter.

시스템에서는 이러한 변조기술 사용시 신호의 복조를

I. 서 론

디지털 유무선통신에서는 BPSK, QPSK, QAM, FSK 등 다양한 변조기술을 사용하고 있다. 통신 시

위해 FIR 필터, 동화기(Equalizer), 정합필터 등을 사용한다^[1-5]. 이때 사용된 정합필터, FIR 필터, 동화기 등은 디지털 회로에 적용시 TDL(Tapped delay line) 필터를 이용한다.

* 아주대학교 전자공학부

* 본 연구는 한국과학재단 연구비지원에 의한 것임(971-0915-095-2)

논문번호: 98357-0814, 접수일자: 1998년 8월 14일

현재 사용중인 TDL 구조들을 살펴보면 QAM 용 상용칩인 Broadcom 사의 BCM3116^[6]에서는 채널 환경의 ISI(Intersymbol Interference)를 줄이기 위한 DFE(Decision FeedBack Equalizer)에 TDL 구조를 사용한다. WLAN 용 상용칩인 Stanford Telecom 사의 STEL-2000A^[7]와 Harris Semiconductor 사의 HSP3824^[8]에서는 정합필터에 TDL 구조를 사용한다.

특히 STEL-2000A와 HSP3824에서 사용하는 정합필터는 direct-form^[7-9] 필터구조로 2-4Mbps의 데이터 처리 속도를 가지며 11배의 칩 확산에 따라 22-44Mcps의 확산데이터를 처리한다. 이때 사용된 정합 필터의 구조는 여러비트의 곱셈과 덧셈을 한 사이클에 처리하기 때문에 고속에는 불리하다. 따라서 고속의 멀티미디어 연산을 위한 파이프라인 방식의 하드웨어 구조 연구가 필요하다. 현재 파이프라인 기법을 사용한 transpose-form 필터^[9-10]는 고속연산이 가능하나 하드웨어의 출력단에 가까울수록 레지스터의 비트 수가 증가하여 VLSI 면적을 많이 차지하는 단점을 가진다.

현재 이동통신 시스템에서는 코드의 동기획득을 위해 I&D(Integrated and Dump) 방식을 사용하는데 이는 정합필터를 이용한 구조보다 초기 동기획득 면에서 성능이 떨어진다[11]. 이때 사용되는 direct-form 구조의 정합필터는 연산속도 문제를 가지므로 멀티미디어 이동통신에 부적합하며 transpose-form 구조의 정합필터는 하드웨어 면적을 크게 차지하여 전력소비 문제를 가지므로 휴대용 이동통신에 부적합해진다. 따라서 본 논문에서는 이들 문제를 해결하기 위해 실제 칩 구현시 효율적인 하드웨어 구조를 제안한다. 제안된 구조를 사용할 경우 다음과 같은 장점을 지닌다. 첫째, direct-form 필터에 비하여 게이트 수는 증가하나 1.5배 이상의 고속 연산을 수행한다. 둘째, transpose-form 필터에 비하여 연산속도는 같으나 게이트 수를 줄일 수 있다. 셋째, TDL 필터가 빠른 응답을 가지므로 필터 출력을 다양한 타이밍 정보에 사용하기 용이하다.

본 논문의 구성은 다음과 같다. 기존 TDL 필터의 하드웨어 알고리즘 및 구조에 대하여 알아보고 제안된 알고리즘을 이용하여 구현된 하드웨어를 기술한다. IV장에서는 성능평가를 서술하고 결론으로 결론을 맺는다.

II. 기존 TDL 필터의 하드웨어 알고리즘 및 구조

본 장에서는 기존의 TDL 필터의 알고리즘과 하드웨어 구조를 서술한다. 우선 TDL 필터의 기본 알고리즘을 분석하고 기존 TDL 필터의 하드웨어 구조들을 조사한다. 동시에 기존 하드웨어 구조의 장단점을 분석한다.

1. 기존의 TDL 필터 알고리즘

TDL 필터는 식 (1)과 같은 일반적인 입출력 관계식을 가진다^[1].

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

식 (1)에서 $h[k]$ 은 PN(Pseudorandom Noise)-시퀀스나 기준 상수(coefficient) 값을 나타내며 $x[n]$ 은 n 번째 입력샘플(input sample)을 나타낸다. 이때 적분 구간은 0부터 $N-1$ 까지이며 이는 TDL 필터의 탭 수가 N 임을 의미한다. $y[n]$ 은 TDL 필터의 n 번째 출력값을 나타낸다. 이러한 입출력 관계식을 갖는 TDL 필터는 디지털 신호검파에 사용되는 정합 필터, FIR 필터, 동화기 등 다양한 필터에 사용되고 있다.

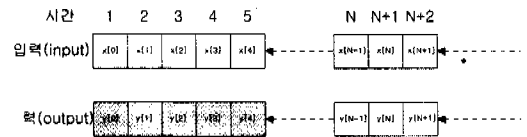


그림 1. TDL 필터의 데이터 흐름도
Fig. 1. The data flow diagram of the TDL filter

그림 1은 TDL 필터의 하드웨어에 대한 데이터 흐름도를 나타낸다. 시간이 N 일 경우 $x[N-1]$ 이 입력되자마자 식 (1)에서 n 이 0, 적분 구간 N 에 대해 연산을 수행하여 하나의 적분 구간 연산이 완료된 $y[N-1]$ 가 출력된다. 이는 입력으로부터 $x[0], x[1], \dots, x[N] \dots$ 순으로 입력샘플이 인가되면 $x[N-1]$ 가 입력 될 때부터 유효한 데이터가 출력되기 시작하여 지속적으로 적분구간의 연산이 완료된 데이터를 출력하기 시작함을 의미한다. 또한 이는 데이터 초기화 시간이 N 임을 의미한다. TDL 필터의 출력을 주파수 오류교정, 위상 오류교정 등의 폐환구조에 응용하려면 그림 1에 나타난 TDL 필터의 데이터 흐름도와 같이 데이터 초기화 시간이 적게 걸려 데이터가 출력 되어야 한다.

일반적으로 연산속도 개선을 위해 연산부의 중간에 레지스터들을 이용하여 임의의 파이프라인 기법을 사용하면 출력이 늦어지므로 폐환 회로의 성능

을 저하시킨다. 따라서 케환회로에 영향이 없는 파이프라인 기법에 대한 연구와 하드웨어 구조가 필요하다.

2. 기존의 TDL 필터 하드웨어 구조

본 절에서는 케환회로에 대하여 영향이 없는 기존의 TDL필터를 위한 수식적인 모델을 제시하고 예를 들어 설명한다. 먼저 TDL 필터를 하드웨어로 쉽게 표현하기 위해 수식적으로 표현한 후 하드웨어 구조로 나타낸다. 하드웨어 구조를 위한 수식으로 식 (1)에서 유추하여 일반적인 direct-form 필터를 표현하면 식 (2)와 같다.

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

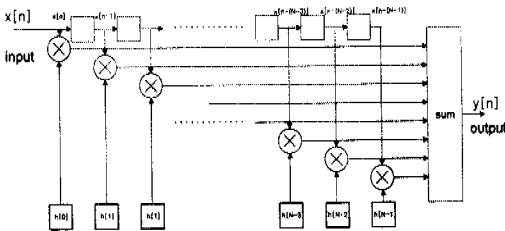


그림 2. Direct-form TDL 필터
Fig. 2 The direct-form TDL filter

이때 Z^{-k} 은 하드웨어 구성에서 레지스터를 나타내며 $x[n]Z^{-k}$ 는 $x[n]$ 입력이 k 개의 레지스터를 거친 $x[n-k]$ 데이터임을 나타낸다. 이는 그림 2와 같이 나타낸다. Direct-form 필터는 그림에서와 같이 여러비트의 덧셈과 곱셈을 한 사이클에 처리해야 하므로 연산 속도가 느린 단점을 가진다. 따라서 고속 연산을 위해 수식에서 $x[n-k]$ 번째 입력 레지스터를 그림 3과 같이 MA(Multiply and Add) 연산 후의 출력 레지스터로 파이프라인 시켜야 한다. 이러한 파이프라인 방법을 변환 수식으로 표현하면 식 (3)과 같다.

$$h[k] \times x[n] Z^{-k} = Z^{-1} (h[k] \times x[n] Z^{-(k-1)}) \quad (3)$$

식 (3)의 변환 수식을 이용할 때 n 은 상수이다. 이때 $h[3] \times x[7] Z^{-3}$ 은 고속연산을 위해 곱셈 연산 후 출력에 레지스터를 사용하면 $Z^{-1} (h[3] \times x[7] Z^{-2})$ 로 바꿀 수 있다. 이는 레지스터 3개를 거친 $x[4]$ 데이터와 $h[3]$ 를 연산하는 것이 레지스터 2개를 거친 $x[5]$ 데이터와 $h[3]$ 연산후 출력에 레지스터를 연결한 것과 같음을 의미한다. 식 (3)을 이용하여 파이

프라인 기법을 사용하면 일반적인 transpose-form 필터의 식 (4)와 같은 하드웨어 수식으로 나타낸다.

$$y[n] = h[0] \times x[n] + Z^{-1} (h[1] \times x[n] + Z^{-1} (h[2] \times x[n] + \dots + Z^{-1} (h[N-2] \times x[n] + Z^{-1} (h[N-1] \times x[n])) \dots)) \quad (4)$$

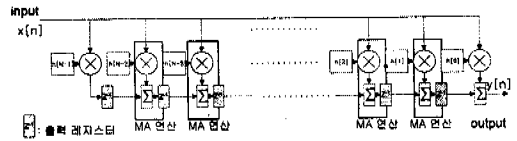


그림 3. Transpose-form TDL 필터
Fig. 3 The transpose-form TDL filter

식 (4)에서 Z^{-1} 은 그림 3에서 출력 레지스터를 나타낸다. transpose-form 필터는 그림에서와 같이 연산속도는 빠르나 하드웨어 구현 시 출력쪽으로 갈수록 레지스터 수가 증가하므로 하드웨어 면적을 많이 차지하고 전력을 많이 소비하는 문제점을 가진다.

III. 제안된 하드웨어 구조

일반적으로 TDL 필터 구조 중에서 케환회로에 영향이 적은 하드웨어 구조로 direct-form 필터와 transpose-form 필터 형태를 사용한다. Direct-form 필터는 필터의 탭 수가 증가하게 되면 연산속도의 문제를 가진다. Transpose-form 필터는 연산속도는 빠르나 구현 시 게이트 수가 증가하여 하드웨어 면적을 많이 차지하고 전력을 많이 소비하는 문제점을 가진다.

이러한 문제점들을 해결하기 위해 본 논문에서 제안한 방법을 이용하면 하드웨어 설계시 direct-form 필터, transpose-form 필터와 다른 형태의 TDL 필터 설계가 가능하며, 연산속도와 하드웨어 면적이 최적화 된다. 따라서 본 장에서는 direct-form 필터, transpose-form 필터와 다른 형태를 가진 TDL 필터중 8탭의 예를 들어 설명한다. 식 (2)의 일반수식을 8탭의 direct-form 필터 하드웨어 수식으로 표현하기 위하여, $N=8$, PN-시퀀스 입력 $n=7$ 을 대입하여 나타낸다. 이는 그림 3의 필터 형태가 되며 식 (5)와 같이 나타낸다.

$$y[7] = h[0] \times x[7] + h[1] \times x[7] Z^{-1} + h[2] \times x[7] Z^{-2} + h[3] \times x[7] Z^{-3} + h[4] \times x[7] Z^{-4} + h[5] \times x[7] Z^{-5} + h[6] \times x[7] Z^{-6} + h[7] \times x[7] Z^{-7}$$

$$xx[7]Z^5+h[6]xx[7]Z^6+h[7]xx[7]Z^7 \quad (5)$$

식 (5)에서 $y[7]$ 을 출력하기 위해 한 사이클에 7개의 덧셈과 8개의 곱셈연산을 수행해야 하므로 고속의 연산이 요구되는 TDL 필터에서는 불리하다. 특히 그 응용이 CDMA 이동통신일 경우 한 심볼당 64칩^[11]이므로 임계경로가 증가하여 연산속도가 더욱 느려지게 된다. 따라서 입력에 사용된 레지스터들 대신 그림 3과 같이 Z^1 의 출력 레지스터를 사용하여 임계경로를 줄일 수 있다.

고속연산을 위해 식 (4)의 일반수식을 8탭의 transpose-form 필터 하드웨어 수식으로 다시 나타내면, 8 탭이므로 $N=8$, PN-시퀀스 입력 $n=7$ 을 대입하여 그림 3의 8탭 TDL 필터가 되며 식 (6)과 같이 나타낸다.

$$y[7]=h[0]xx[7]+Z^1(h[1]xx[7]+Z^1(h[2]xx[7]+Z^1(h[3]xx[7]+Z^1(h[4]xx[7]+Z^1(h[5]xx[7]+Z^1(h[6]xx[7]+Z^1(h[7]xx[7])))))) \quad (6)$$

식 (6)은 임계경로를 줄이기 위한 그림 3의 출력 레지스터를 MA 출력에 사용하여 나타낸 것이다. 식 (6)에 나타나듯이 Z^1 은 덧셈기와 곱셈기를 부분적으로 나누는 파이프라인 역할을 수행한다. 이때 그림 3의 출력 레지스터는 최종 출력에 가까워질수록 비트 수가 증가한다. 따라서 하드웨어 부담을 줄이기 위해 그림 3에서 $N=8$ 일 때 우측의 MA 연산에 대한 임계경로를 고려하여 게이트 수를 줄이기 위한 부분합으로 나타낼 수 있다. 이때 사용되는 부분합은 여러비트의 입력을 동시에 더하는 연산을 수행한다. 따라서 본 연구에서는 이러한 관계식을 이용하여 VLSI 면적과 속도를 고려한 TDL 필터를 제안한다.

사용된 필터는 4비트 입력데이터에 8탭 연산을 하는 TDL 필터로 가정한다. 이때 그림 4에서 A 레지스터의 출력과 $h[0]$ 의 곱셈 출력과의 합 연산은 $h[1]$ 의 MA 출력 데이터 7비트와 입력데이터 4비트 덧셈으로 ripple adder 사용시 XOR 7단을 거친다. 이러한 최대 임계경로를 고려하면 최대 7단의 XOR를 거치는 여러 랩들의 출력을 4비트나 6비트를 이용하여 동시에 더하는 덧셈기(adder tree)로 나타낼 수 있다. 이는 8탭 TDL 필터에서는 그림 4의 D 출력레지스터의 입력전인 덧셈기 {4+4+4}, 그림 4의 C 출력레지스터의 입력전인 덧셈기 {6+4+4}로 나타낸다. 이때 4, 6은 입력 데이터의 비트 수를 나

타낸다. 덧셈기 {4+4+4}, {6+4+4}는 adder tree를 사용하면 {4+7}의 ripple adder와 비슷한 임계경로를 가진다. 이때 사용된 덧셈기는 양수의 사용시 나타낸 것이다. 만일 음수를 포함한 덧셈기를 사용할 경우 sign 확장만 하면 되므로 레지스터 수나, 임계경로는 거의 같다. 이러한 부분합들을 이용하여 수식적으로 표현하면 식 (7)과 같다.

$$y[7]=h[0]xx[7]+Z^1(h[1]xx[7]+Z^1(h[2]xx[7]+Z^1(h[3]xx[7]+h[4]xx[7]Z^1+Z^1(h[5]xx[7]Z^1+h[6]xx[7]Z^2+h[7]xx[7]Z^3)))) \quad (7)$$

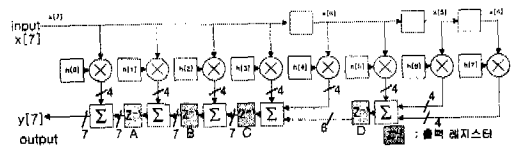


그림 4. 제안된 TDL 필터
Fig. 4 The proposed TDL filter

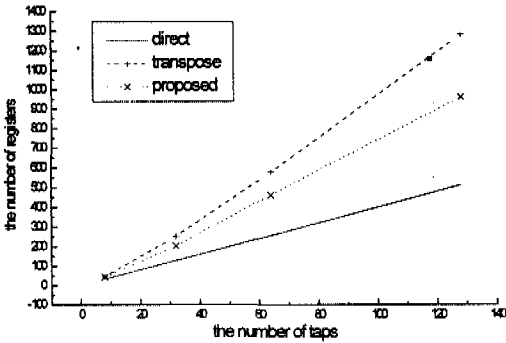
식 (7)의 유도 과정은 다음과 같다. 먼저 식 (2), (4)로부터 탭 수에 맞는 N, n 을 대입하여 식 (5), (6)을 구한다. 그 다음 탭 수에 따른 임계경로를 구하며 그로부터 임계경로를 고려하여 동시에 더할 수 있는 부분 집합을 구한다. 여기서 8탭의 부분 집합은 {4+4+4}, {6+4+4}이다. 식 (5), (6)에서 4비트 입력 3개, 즉 {4+4+4}를 동시에 더하는 구조에서 낮은 순으로 각각의 원소인 $h[k]$ 값을 선택한다. 이는 수식에서 높은 순서인 $h[7], h[6], h[5]$ 3개를 나타낸다. 그 다음 3개의 원소를 식 (5)의 direct-form 필터 구조로 더하도록 한 다음 출력에는 Z^1 을 붙인다. 식 (7)에서는 괄호안에 나타난 $h[5]xx[7]Z^1, h[6]xx[7]Z^2, h[7]xx[7]Z^3$ 의 덧셈이며 이는 출력에서 6비트 데이터가 된다. 그림 4에서는 D 출력레지스터 전단의 3개 입력을 연산하는 덧셈기이다. 또한 {4+4+6}은 그 다음 높은 순서인 $h[3], h[4], (h[5], h[6], h[7])$ 을 선택하며 이들을 direct-form 필터 구조로 더한다. 이는 식 (7)에서 $h[3]xx[7], h[4]xx[7]Z^1, Z^1(h[5]xx[7]Z^1+h[6]xx[7]Z^2+h[7]xx[7]Z^3)$ 의 덧셈이며 그림 4에서는 C 출력레지스터 전단의 3개 입력을 연산하는 덧셈기이다. 그 다음 식 (3)을 이용하여 식 (5), (6)으로부터 식 (7)을 유도한다. 식 (7)을 하드웨어 구조로 나타내면 그림 4와 같이 된다. 따라서 제안된 필터는 출력레지스터와 입력 레지스터와의 관계를 이용

하여 하드웨어 수식을 사용하므로 출력 응답 지연이 없게 된다. 이렇게 함으로써 $x/7$ 이 입력되면 바로 $y/7$ 이 출력으로 나와 변환 구조를 요구하는 곳에 TDL 필터 출력을 이용할 수 있다.

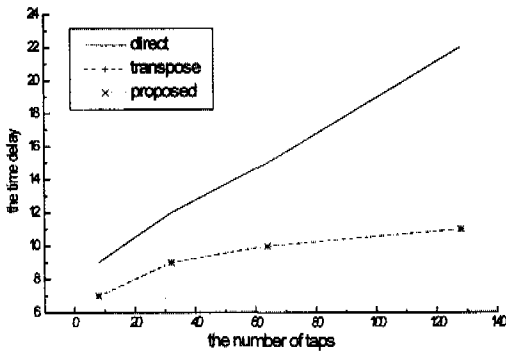
제안된 구조는 8탭의 경우를 나타낸 것이다. 8탭 이상의 경우 식 (2), (3), (4)를 이용하여 파이프라인 기법을 사용하면 데이터 초기화 시간이 direct-form 필터, transpose-form 필터와 같은 하드웨어 구조가 개발된다. 이는 direct-form 필터보다 고속의 연산이 가능하며 동시에 transpose-form 필터보다 게이트 수가 적게되는 장점을 가진다.

IV. 성능 평가

본 장은 direct-form 필터, transpose-form 필터, 제안하는 구조의 성능 평가를 나타낸다.



(a) 레지스터수 비교
(a) The comparison of the number of registers



(b) 시간지연 비교
(b) The comparison of the time delay

그림 5. 성능 분석
Fig. 5 The performance comparison

그림 5는 4비트 입력 데이터에서 탭 수 변화에 따른 레지스터의 수와 시간지연을 비교한 것이다. 그림 5(b)에서 사용된 시간은 TDL 필터에 사용된 덧셈기의 시간 지연(time delay)을 나타낸다. 덧셈기는 ripple adder를 고려한 것으로 T_a 는 full adder 한 개의 시간 지연을 나타낸다. 따라서 N_b 비트 ripple adder의 경우 $N_b T_a$ 의 시간지연을 갖게 된다. 제안한 구조는 임계경로를 고려한 부분 합 연산을 수행하므로 레지스터가 43개로 transpose-form 필터의 46개에 비하여 9% 줄어들고 임계 경로는 같은 $7T_a$ 이므로 연산 속도는 같다. 또한 direct-form 필터의 32개 레지스터 수보다 증가하나 속도는 4비트 8개를 동시에 더하는 것을 파이프라인 하여 4비트와 7비트를 더하므로, $9T_a$ 에서 $7T_a$ 로 줄어들어 1.5배 이상 빠른 연산을 수행하는 장점을 지닌다. 실제 CDMA 디지털 이동통신 표준인 IS-95의 경우 심플당 칩수가 64이므로 이를 2번 샘플링 하게 되면 정합필터의 탭 수가 128이므로 이를 transpose-form 필터와 비교하면 레지스터 수가 1281개에서 960개로 약 30%이상 줄일 수 있다. 심플당 칩수가 늘어날수록 차이는 더욱 커져 하드웨어를 많이 절약할 수 있다.

V. 결론

본 논문에서 제안된 TDL 필터는 하드웨어 면적과 연산속도의 상호 균형(trade-off)을 고려한 설계이다. 기존 direct-form 필터는 탭 수가 크면 임계 경로가 커서 저속으로 동작한다. 고속 동작을 위한 transpose-form 필터는 하드웨어가 커서 전력을 많이 소비하는 문제점을 가진다. 본 논문에서 제안된 TDL 필터는 direct-form 필터에 비하여 게이트 수는 증가하나 연산속도는 고속으로 동작하며, transpose-form에 비하여 연산속도는 같으나 게이트 수는 줄어드는 장점을 가진다. 또한 본 설계 방법을 이용하여 고속이 요구되는 응용에서는 파이프라인 레지스터를 많이 삽입하여 임계경로를 줄여 연산속도를 높일 수 있고, 전력소비가 적게 요구되는 저속 응용에서는 출력의 파이프라인 레지스터를 적게 삽입하여 VLSI 면적을 줄일 수 있다. 따라서 제안된 알고리즘을 사용하여 속도에 맞는 최적의 TDL 필터를 찾아낼 수 있다. 본 논문에서 제안된 정합필터의 최적화 알고리즘은 BPSK, QPSK, QAM 등 디지털 통신의 신호 검파를 위한 정합필터, 채널 환경의 ISI를 줄이기 위한 등화기, FIR 필터 등에 사용

할 수 있다.

참 고 문 헌

[1] E. C. Ifeachor, B. W. Jervis, Digital Signal Processing, Addison-Wesley, 1996.

[2] J. G. Proakis, Digital Communications, McGRAW-HILL, 1995.

[3] M. D. Yacoub, Foundations of Mobile Radio Engineering, CRC Press. 1993

[4] G. L. Stuber, Principles of Mobile Communication, Kluwer Academic publishers. 1996.

[5] S. G. Wilson, Digital Modulation And Coding, Prentice Hall. 1996.

[6] Broadcom Co., BCM3116, 1997

[7] Stanford Telecom Inc., The Spread Spectrum Handbook, 1994.

[8] Harris Semiconductor Co., HSP3824 Preliminary Data Sheet, 1995.

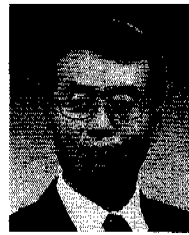
[9] R. Jain, R. T. Yang, T. Yoshino, "FIR-GEN: A Computer-Aided Design System for High Performance FIR Filter integrated circuit," *IEEE Trans. Signal process.*, vol. 39, pp. 1655-1668, July 1991.

[10] 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 μ m BiCMOS gate array," *IEEE J. Solid-State Circuit.*, vol. 25, pp. 1494-1501, Dec. 1990.

[11] A. Polydoros. C. L. Weber, "A Unified Approach to Serial Search Spread-Spectrum Code Acquisition-Part II : A Matched-Filter Receiver," *IEEE Trans. on Comm.* vol. 32, no. 5 pp. 550-560, May 1984.

선우 명훈(Myung Hoon Sunwoo)

정회원



1980. 2 : 서강대학교 전자공학 학사

1982. 2 : 한국과학기술원 전자공학 석사

1982. 3~1985. 8 : 한국전자통신연구소(ETRI) 연구원

1985. 9~1990. 8 : Univ. of Texas at Austin 전자공학 박사

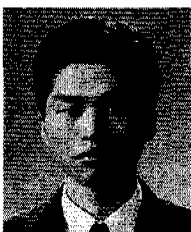
1990. 8~1992. 8 : Motorola, DSP Chip Division, 미국

1992. 8~1996. 10 : 아주대학교 전기전자공학부 조교수

1996. 10~현재 : 아주대학교 전기전자공학부 부교수 <주관심 분야> VLSI 및 Parallel Architecture, 통신, 영상 및 신호처리용 ASIC 설계

서 승 완(Seung Wan Seo)

정회원



1997. 2 : 아주대학교 전자공학 학사졸업(공학사)

1997. 2~현재 : 아주대학교 전자공학 석사과정

<주관심 분야> 영상, 통신 및 신호처리용 ASIC 설계