

재구성 가능한 FIR 필터 하드웨어 구조 설계

Design of Reconfigurable Hardware for FIR Filters

동성수*, 이종호**

(Sung Soo Dong*, Chong Ho Lee**)

Abstract - In general, for specific applications, customized hardware showed better performance than general processor in terms of processing time and power consumption. However, customized hardware systems have lacks of flexibility in nature and it leads the difficulties for debugging and architecture level revision for performance enhancement. To solve this problem, reconfigurable hardware is developed. Proposed reconfigurable hardware architecture for FIR filter system can easily change the architecture of filter blocks including filter tap size and their signal path. Proposed FIR filter architecture was implemented on FPGA using several MUXs and registers and it showed the reconfigurability and reusability in several examples.

Key Words :Reconfigurable hardware, FIR filter, FPGA implementation

1. 장 서 론

기술적인 측면에서 바라보았을 때, 주변의 많은 사물 들은 어떤 목적을 위해 제어되는 대상물체이다. 특히 각종 정보통신 및 가전제품 들은 프로세서 등을 통해 제어된다. 이는 소프트웨어 방식으로 처리하는 것이 설계시간 및 수정보완 측면에서 쉬운 해결책 중 하나이기 때문이다. 그러나 특정의 응용에서는 하드웨어 방식에 비해 소프트웨어 방식이 효율과 전력 소모 면에서 불리하다. 따라서 전용의 하드웨어를 사용하면 그러한 문제를 해결할 수 있지만, 하드웨어 특성상 내부 구조를 변경 하여야 하는 경우는 어려움이 있다. 약간의 구조 수정을 하려고 하는 경우도 하드웨어를 다시 만들어야 하기 때문에 시간과 비용이 많이 든다. 그러므로 재구성 하드웨어가 필요하다[1,2]. 재구성 하드웨어는 목적하는 기능의 성능개선이나 수정보완을 위해서 빠른 시간 내에 하드웨어 구조를 변경할 수 있다는 장점을 가진 전용 하드웨어로 동작할 수 있다. 본 연구는 특정 용도의 하드웨어 블록에 대해서 재구성 가능한 구조를 제안하고, 그 유용성을 확인 하는데 목적을 두고 있다. 대상 블록을 먼저 FIR(Finite Impulse Response) 필터로 선정 하였다. 아날로그 필터에 비해 디지털 필터는 성능, 잡음특성, 구현의 용이성 등으로 인해 많이 사용된다. 특히 FIR 필터는 IIR(Infinite Impulse Response) 필터 대비, 같은 주파수 특성을 얻기 위해 높은 차수가 필요하다는 단점이 있지만 안정도, 선형 위상, 구현의 편의성 등

으로 인해 널리 이용된다. 따라서 재구성 하드웨어 연구를 위한 블록으로 많은 응용 분야에서 사용되고 있는 FIR 디지털 필터를 먼저 선정하여 연구하였다[3,4]. FIR 필터는 응용에 따라서 탭 수가 가변 된다. 따라서 탭 수가 고정되어 있는 하드웨어인 경우, 응용에 따라 적은 탭 수를 필요로 할 때는 잉여 탭의 계수에 영(0)을 적용하여 연산한다. 그러나 이는 불필요한 연산과 대기시간(latency)을 초래 하므로 효율이 떨어진다. 이런 문제점을 극복하기 위해 가변 탭 구조의 필터가 여러 응용 분야에서 많이 쓰인다. 하지만 이런 구조도 사용하지 않는 부분의 탭 들은 잉여 자원이 되는 문제가 있다. 본 논문에서는 재구성 가능한 하드웨어 구조를 FIR 필터를 대상으로 제안하였고, 탭 수를 가변 할 수 있도록 하여 연산 효율을 높일 수 있게 함은 물론이고 사용되지 않는 부분의 자원들은 또 다른 채널의 FIR 필터로 구성할 수 있도록 하여 하드웨어 자원의 활용도도 높일 수 있게 하였다. 논문은, 2장에서 본 연구의 참조가 되는 FIR 필터에 대해 언급하고, 3장에서는 제안된 재구성 가능한 구조에 대해 논하며, 4장에서는 실험 결과에 대해 살펴보고, 5장에서 결론을 맺는 내용으로 구성하였다.

2. 장 FIR 필터 모델

FIR 디지털 필터의 차분방정식 및 전달함수는 식 (1) 과 식 (2) 로 나타낸다.

$$y(nT) = \sum_{k=0}^M a_k x(nT - kT) \quad (1)$$

$$H(z) = \sum_{n=0}^M a_n z^{-n} = \sum_{n=0}^M h(nT) z^{-n} \quad (2)$$

저자 소개

*龍仁松潭大學 디지털情報科 助教授

**仁荷大學校 情報通信工學府 教授 · 工博

데이터의 처리량을 극대화시키기 위해서 병렬 처리 방식을 고려하고, 유연한 하드웨어 구조를 가지기 위해서 직접형 구성법을 기초로 하였는데, 이는 그림 1에 나타난 구조와 같다.

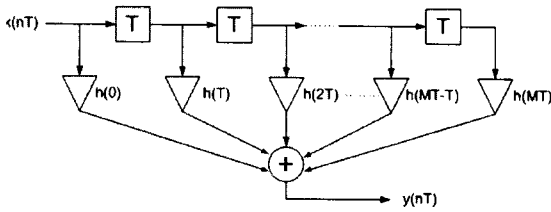


그림 1. 직접형 구조

식 (2)의 전달함수 $H(z)$ 를 식 (3)과 같이 변형할 수 있다.

$$H(z) = h(0) + z^{-1}[h(T) + \dots + z^{-1}[h(MT-T) + z^{-1}h(MT)]] \quad (3)$$

이러한 구성을 전치형 구조라 하며, 직접형 구조에서 입력과 출력을 서로 바꿔놓은 형태이다. 입력 데이터의 전송경로에 지연소자를 주는 대신 누산기의 경로에 지연소자를 사용함으로써 파이프라인 효과 및 구조 단순화에 도움을 준다[5]. 그림 2에 그 구조를 나타내었다.

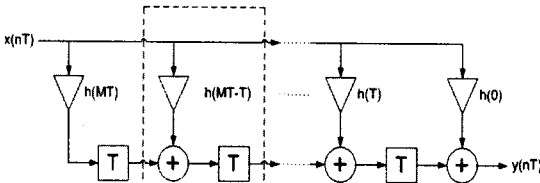


그림 2. 전치형 구조

재구성형 하드웨어로는 단순하면서 규칙적인 구조가 모듈 형식으로 블록화 하기에 적합하기 때문에 전치형 구조를 기본으로 하여 연구하였다.

3. 장 제안된 재구성형 구조

그림 2에서 점선으로 표시한 블록(곱셈기, 덧셈기, 지연소자로 구성)을 기본적인 재구성 모듈(Reconfigurable Module: RM)로 하여 그림 3과 같은 구조를 제안 하였다.

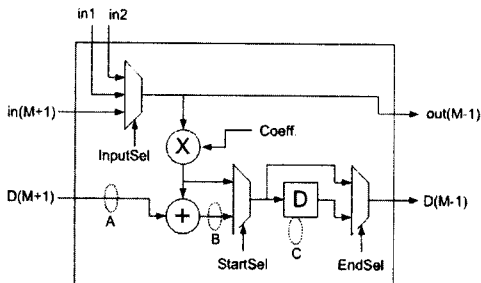


그림 3. 제안된 재구성형 구조(RM)

그림 3의 왼쪽 상단은 들어오는 입력(x(nT))을 선택하게 만든 부분이다. 필터의 처음 블록으로 사용될 때는 입력들 (in1, in2) 중 하나를 선택하고, 필터의 중간이나 마지막 블록으로 사용될 때는 옆 블록에서 들어오는 신호(in(M+1))를 사용할 수 있게 구성하였다. 따라서 RM은 필요시 여러 채널의 필터에 사용될 수 있고 입력채널 개수는 응용에 따라 가변하면 된다. 그림 3의 중앙 하단 부는 RM이 필터의 처음 블록으로 사용될 때 곱셈기의 출력을, 중간이나 마지막 블록으로 사용 시에는 덧셈기의 출력을 사용할 수 있도록 설계한 부분이다. 그림 3의 우측 하단 부는 RM이 필터의 마지막 블록으로 사용될 때 지연소자를 거치지 않는 경우와 처음 또는 중간 블록으로 사용되어 지연소자를 거치는 경우를 선택하도록 구성한 부분이다.

부가적으로 언급하자면, HDL(Hardware Description language)로 설계하여 회로 합성을 하는 경우 무 입력 (floating input) 오류를 막기 위해 그림3의 좌측 하단 부의 A 부분에 멀티플렉서로 영(0)입력 처리한다. B 부분은 덧셈 결과의 비트 차이를 보완해 주는 반올림, 절단, 버림 등의 회로를 더 구성할 수 있고, C 부분은 지연소자의 클럭에 멀티플렉서 처리하여 필터 채널 마다 다른 샘플링 주파수를 이용할 수 있도록 처리하는 부분이다.

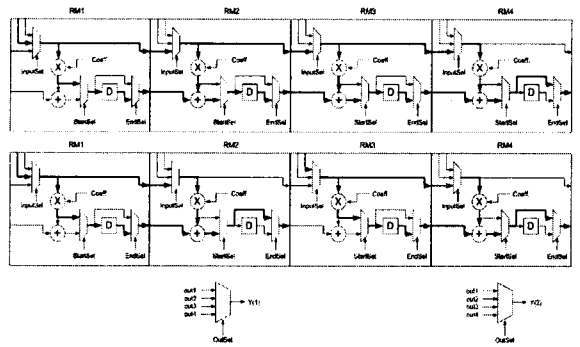


그림 4. 재구성형 구조를 이용한 FIR 필터 설계 예

제안된 구조를 이용하여 FIR 필터를 구성한 예제를 그림 4에 보였다. RM을 이용한 구성 예를 보이기 위해서 간단한 탭 수를 이용하였다. 그림 4의 상단의 부분은 RM 4개를 이용하여 4탭의 FIR 필터 구성을 보인 것이고, 중간은 RM 4개를 이용하여 2채널의 2탭 FIR 필터 구성을 보인 것이다. 하단의 멀티플렉서는 출력(y(nT))을 나타내는데, 입력의 채널 개수만큼 구성하면 된다. 그림의 예제는 짝수 개 탭의 필터 구성을 보였는데, RM이 짝수 개로 구성된 상태에서 홀수의 필터 탭을 필요로 할 때나 그 반대의 경우, 또는 내장한 RM을 사용하지 않는 경우는 연산에 영향을 안 주는 동작을 해야 한다. 그림 3에 표시는 하지 않았지만, 그림3 좌측 상단의 입력 선택 부에 영(0)입력을 더 넣어 연산에 영향을 주지 않도록 할 수 있고, 그림 3의 우측 하단의 C 부분에 샘플링 주파수 선택부에도 멀티플렉서를 구성할 때 영(0)입력을 더 넣어서 RM을 사용하지 않을 경우에 클럭 동작을 하지 않도록 하여 소모전류도 줄일 수 있다.

4. 장 실험 방법 및 결과

재구성 가능한 블록을 FIR 필터를 대상으로 하여 연구한 구조가 제안된 RM이다. 이 구조를 VHDL로 설계 후, Xilinx ISE Tool을 사용하여 회로합성을 하였다. 게이트 수는 각 16비트인 2입력의 곱셈기가 1577개, 각 32비트인 2입력의 덧셈기가 378개 이다. 가변구조를 위한 멀티플렉서를 각 16비트의 입력 4개로 고려했을 때 240개의 게이트 수를 가진다. 나머지 작은 부분들을 고려하면 기존의 전치형 FIR 블록 보다 RM을 구성하기 위해서는 약 20% 정도의 하드웨어 자원의 오버헤드가 있다. 그러나 8탭의 구조를 가진 FIR 필터인 경우에 단지 4탭의 FIR 필터만 사용하는 경우와 4탭의 FIR 필터 2개를 사용할 수 있는 경우를 생각하면 충분히 오버헤드를 극복할 수 있는 장점이 있다. 실제 필터 특성을 보기 위해 16탭의 FIR 필터를 구성하여 Matlab을 이용하여 실험을 하였다. 같은 조건에서 floating-point 로 실험한 경우와 RM으로 구성하여 특성을 비교한 내용을 그림 5에 나타내었다.

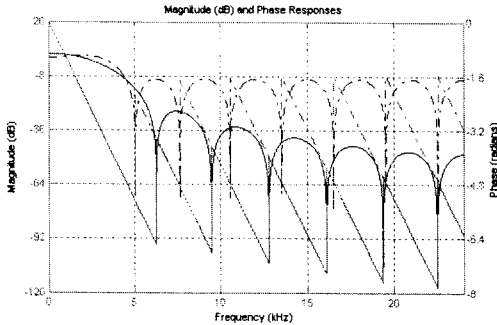


그림 5. 기본 FIR 필터와 RM과의 주파수 특성 비교

그림 5에서 점선으로 표시된 부분이 floating-point의 특성이고, 실선이 RM으로 같은 내용의 FIR 필터를 구성하여 나타낸 특성이다. 오차가 나는 것을 볼 수 있으나 FIR 필터이므로 사용가능한 범위에 들어온다. 좀 더 정밀한 분석을 위해서 더 많은 실험을 진행 중이다.

5. 장 결론

특정용도에 있어서 소프트웨어를 이용하는 범용의 프로세서 보다 전용의 하드웨어가 프로세싱 속도나 전력 소모 측면 등에서 더 효율적이다. 하지만 하드웨어는 특성상 구조가 한번 정해져서 제작이 되면 작은 오류라도 고치기가 어렵고 성능 향상을 위해 구조를 변경하고 싶은 경우라도 적용하기가 쉽지 않다. 새로 제작하는데 시간과 비용이 많이 들기 때문이다. 따라서 목적블록의 내부 구조를 쉽게 변경할 수 있는 재구성형 하드웨어가 필요하다. FPGA는 재구성형 하드웨어이지만 범용이므로 특정 용도에 있어서는 전용의 하드웨어를 이용하는 것보다 효율이 떨어진다. 그러므로 전용의 하드웨어 능력에 가까우면서도 재구성이 가능한 가변구조의 하드웨어를 연구하는 것은 의미 있는 일이라 여겨진다. 본 연구에서는 응용을 FIR 필터로 하여 필터 구조에 대한 재구성형 구

조를 제안하여 실험하였다. 실험결과 재구성형 구조가 응용에 따라 가변 하는 경우에 하드웨어 사용면에서 효율적이라는 가능성을 보였다. 앞으로 재구성형 하드웨어를 적용하는 응용블록을 더 늘리고, 재구성형 하드웨어를 플랫폼으로 하여 신경망 알고리즘, 유전알고리즘 등을 이용하여 진화 적응형 하드웨어를 연구하려고 한다.

참 고 문 헌

- [1] M. Wirthlin and B. Hutchings, "A Dynamic Instruction Set Computer", IEEE Symposium on FPGAs for Custom Computing Machines, pp. 99-107, 1995.
- [2] P. French and R. Taylor, "A Self-Reconfiguring Processor", IEEE Symposium on FPGAs for Custom Computing Machines, pp. 50-59, 1993.
- [3] S. Mitra and J. Kaiser, *Handbook Digital Signal Processing*, Wiley, 1993.
- [4] J. Proakis and D. Manolakis, *Digital Signal Processing: Principles, Algorithms, & Applications*, 1996.
- [5] J. Johnson, *Introduction to Digital Signal Processing*, Prentice-Hall, 1989.