

양준석, 김기철

서울시립대학교 전자전기컴퓨터공학부

fpga@uos.ac.kr, kkim@uos.ac.kr

A Design of Memory Efficient Pipelined FFT Processor

Joonseok Yang, Kichul Kim

Department of Electrical and Computer Engineering

University of Seoul

요 약

본 논문에서는 전치메모리와 서플메모리를 사용하여 효율적인 메모리 구조를 갖는 파이프라인 방식의 FFT 프로세서를 제안한다. 기존의 파이프라인 방식의 FFT 프로세서는 연산속도는 빠르다 요구되는 메모리의 크기가 크고 계산할 수 있는 FFT의 크기가 제한되는 단점이 있다. 제안된 구조에서는 전치 메모리와 서플 메모리를 적절히 조합하여 주어진 크기의 FFT 프로세서가 최소의 메모리를 사용하면서 데이터의 정체 없이 연속적으로 연산을 수행할 수 있도록 하였다. 본 논문에서는 제안된 구조로 2의, $4 < n < 14$, 크기를 갖는 파이프라인 방식의 FFT들에 대하여 최소한의 메모리를 가지고 구성하는 방법과 구성된 각 FFT가 요구하는 메모리의 크기를 보인다.

1. 서론

FFT(Fast Fourier Transform) 프로세서는 여러 신호처리 알고리즘의 기본으로서 차세대 디지털 통신, 신호처리, 멀티미디어 기기의 핵심 소자로 이용되며 하드웨어의 복잡도도 매우 높다. 일반적으로 FFT 프로세서를 구성하는 방식으로 pipelined 방식과 shared memory(memory based) 방식이 사용된다[1].

Pipelined 방식은 보통 radix 2, radix 4 방식에 기반한 butterfly 유닛과 메모리, 곱셈기 등을 직렬로 연결하여 연속적으로 데이터를 처리한다[2,3]. 따라서 높은 처리속도를 갖지만 각 단계 들어가는 메모리, 곱셈기, 덧 등의 크기 때문에 전체 칩의 크기가 현저하게 커지는 단점이 있다 [1]. 이와 같이 메모리의 크기가 커지는 단점을 극복하기 위한 많은 연구들이 수행되고 있다. Lenart와 Owall은 각 단계에서 생성되는 중간값들을 저장할 때 최대값을 기준으로 크기를 잘라서 저장하는 convergent block floating point(CBFP) 방식을 사용하여 메모리의 크기를 줄이려 했으나 Khashab과 Swartzlander는 각 단계의 프로세스 엘리먼트들을 모듈로 곱모듈 사이에서 중간 메모리를 써서 전체 메모리의 크기를 줄이려 했다[2,3].

Shared memory 방식은 사용되는 전체 메모리의 크기를 줄이기 위해 메모리와 butterfly 유닛, 곱셈기 등을 한 개 또는 적은 양을 사용해서 반복해서 사용하면서 연산을 수행한다[4,5]. 그러므로 전체 칩의 크기는 작아지지만 데이터의 처리속도가 떨어지게 된다[1]. 따라서 메모리의 크기는 그대로 유지하면서 속도를 향상시키기 위한 많은 연구

가 수행되고 있다. Radhouane과 Liu, Modlin은 메모리의 상태와 FFT의 모드를 바꿔가면서 연산을 수행하는 continuous flow mixed mode(CFMM)를 사용하여 성능을 높이려 했고 Ma와 Wanhammar는 메모리를 네 개로 나누고 butterfly 연산의 순서를 조절해서 연산량을 최소한으로 줄이려 했다[4,5].

본 논문에서는 풀리-투키 알고리즘을 사용하여 pipelined 방식의 FFT를 구성한다. 설계해약한 1-D FFT를 풀리-투키 알고리즘을 사용하여 두 개의 작은 FFT로 나누고 중간값을 저장하는 메모리와 트위틀 팩터 곱셈기를 이용하여 행렬분리변환방법으로 데이터를 연속적으로 처리한다. 이 방법에서 일반적으로 중간값을 저장하는 메모리로 전치 메모리가 많이 사용된다. 하지만 전치 메모리는 풀리-투키 알고리즘을 이용한 $M \times N$ FFT의 구현에서 M 과 N 이 같은 경우에만 사용할 수 있기 때문에 위의 구조를 모는 크기의 FFT에 적용하기 위해서는 M 과 N 의 크기가 다를 경우에도 사용할 수 있는 메모리가 필요하다. 이를 해결하기 위해 본 논문에서는 M 과 N 이 다른 경우에도 사용할 수 있는 새로운 형태의 메모리인 서플 메모리와 전치 메모리를 함께 사용하여 메모리의 낭비가 없는 효율적인 FFT의 구현을 보인다. 또한 제안된 방법으로 구성할 수 있는 가장 효율적인 메모리 구성 및 크기를 각 FFT 크기별로 제안한다.

본 절의 서론에 이어 제 2절에서는 풀리-투키 방법을 사용한 구조와 사용되는 메모리 시스템에 대해 소개한다. 제 3절에서는 전치 메모리와 서플 메모리를 사용한 FFT의 구조 및 구현에 대해 소개하고 pipelined로 설계된 다른