

시스템릭 아키텍처를 갖는 FFT 프로세서의 설계 *

○ 강병훈, 정성욱, 이장규, 최병운, 신경욱, 이문기

연세 대학교 전자 공학과

Design of FFT processor with systolic architecture

B.H.Kang, S.W.Jeong, J.K.Lee, E.Y.Choi, K.W.Shin and M.K.Lee

Dept. of Electronics Eng., Yonsei Univ.

ABSTRACT

This paper describes 16-point FFT processor using systolic array and its implementation into VLSI. Designed FFT processor executes FFT/IFFT arithmetic under mode control and consists of cell array, array controller and input/output buffer memory. For design for testability, we added built-in self test circuit into designed FFT processor.

To verify designed 16-point FFT processor, logic simulation was performed by YSILOG on MICRO-VAXII.

From the simulation results, it is estimated that the proposed FFT processor can perform 16-point FFT in about 4400[ns].

1. 서론

최근에 반도체 기술의 발달로 단일 칩안에 수십 만개의 소자를 집적화하는 것이 가능하게 됨에 따라, 고속 신호처리를 위한 전용 프로세서에 대한 연구가 활발히 진행되고 있다.

FFT (Fast Fourier Transform) 은, 디지털 신호처리 분야에서 대표적인 알고리즘으로서, [2] 영상처리, 통신등 광범위한 분야에 응용이 되고 있다. 정보화 시대가 대두됨에 따라 광대한 데이터를 실시간으로 처리하는 것이 요구되나, 기존 FFT 프로세서는 대수본 소프트웨어 기법을 사용하기 때문에 실시간(real time)처리가 불가능하다. [6] 따라서 다수개의 프로세서를 사용하는 연구가 널리 연구되고 있다 [3]-[5].

본 연구는 최근에 각광을 받고 있는 시스템릭 아키텍처를 사용하여 [7], 16-point FFT 프로세서를 구현하고, 논리 시뮬레이션을 통하여 설계된 회로가 바람직하게 동작함을 확인하였다.

설계된 칩의 테스트를 용이하게 하기 위해서, built-in self test 하드웨어를 추가하였다.

2. 시스템릭 어레이상에 구현된 FFT 알고리즘

N-point FFT 연산은 두개의 데이터 A, B에 대한 식 (1)과 같은 버터플라이 연산을 기본으로 하여 $\log_2 N$ stages 에 걸쳐 계산된다. [2]

$$X = A + B \cdot W^{nk} \quad \text{---a) (1)}$$

$$Y = A - B \cdot W^{nk} \quad \text{---b)}$$

$$n, k = 0, 1, 2, \dots, N-1$$

$$W = \exp(-j2\pi/N); \text{ twiddle factor}$$

그림(1)은 N=16인 경우의 FFT 연산 흐름도이다. 각 연산 stage에서, 식(1)의 버터플라이 연산이 이루어지기 위해서는 데이터 사이에 배열을 위한 곱

셈이 필요하다.

한편 N개의 주파수 스펙트럼 F(n) 에 대한 IDFT (Inverse Discrete Fourier Transform)은 다음과

같이 정의된다. [5]

$$f(k) = \frac{1}{N} \left(\sum_{n=0}^{N-1} F^*(n) W^{nk} \right)^* \quad (2)$$

따라서 IDFT를 계산하기 위해서는, FFT 알고리즘을 그대로 적용할 수 있다. 즉 FFT 알고리즘에서 입력 데이터의 공액복소수로 만들고, $1/N$ 만큼 scale down시킨 형태이다.

N개의 데이터 표본값을 FFT 연산하기 위한 2차원 시스템릭 어레이는 그림(2)와 같이 N개의 셀을 정방형으로 배열하여 구성되며, 데이터는 매 stage마다 상, 하 또는 좌, 우의 인접한 셀을 통해서 연산이 이루어지게 된다.

기존의 연구에 의해서, 매 stage마다 각각의 셀에서는 식(1)의 (a), (b) 중 하나만을 수행하는 그림(3)과 같은 반쪽 버터플라이 연산(HBA: half butterfly arithmetic)이 필요하다. 그림에서 □ 연산자는 "+" 또는 "-" 연산을 의미한다. [3]-[5]

3. VLSI로 구현된 FFT 하드웨어

N개의 셀로 구성되는 2차원 어레이 프로세서는, 집적도의 한계 때문에 N이 작은 경우에만 어레이 전체를 단일 칩속에 집적시킬 수 있게 된다. 그리고 셀간의 데이터 이동이 bit-parallel로 이루어지는 경우 I/O 핀수는 칩에 집적되는 셀의 갯수에 비례해서 증가한다. 이와 같은 단점을 고려하여, N=16으로 고정시키고 데이터 I/O시 실수항과 허수항을 분할하여 순차적으로 보내는 방식을 채택하여 핀수의 제약문제를 해결하였다. [8]-[9]

또한 회절인자(twiddle factor) look-up table을 각각의 셀에 두었으며, 매 stage마다 각각의 셀에 필요한 반쪽 버터플라이 연산 형식에 따라 제어되는 데이터 배열을 효율적으로 수행하기 위해서, 셀 식별 숫자(CIN: Cell identification number)를 각각의 셀에, 할당하였다. N=16인 경우, CIN은 4비트로서, 그림(2)의 셀 index와 동일하며 매 stage에서 셀에 필요한 HBA의 형식은 CIN의 MSB비트에서 LSB비트로 순차적으로 선택된다.

N=16인 경우 FFT 프로세서는 그림(4)와 같다. Host 프로세서는 임-출력 버퍼 메모리를 제어하는 것 외에도, FFT 프로세서와 인터페이스되어 정규의(normal) FFT 동작과 함께 FFT 프로세서의 테스트 과정도 제어한다.

입, 출력 버퍼 메모리는 FIFO(First-In First-Out) 특성을 갖는 쉬프트 레지스터 어레이(shift register array)로 구성되어, parallel로 데이터를 입, 출력시켜 I/O소요시간을 최소화한다. 단 I/O시 핀수의 제약을 고려하여, MUX(multiplexer)가 셀 어레이의 입, 출력에 놓여, 실수항과 허수항을 순차적으로 보내도록 제어한다. 그리고 공액복소기(conjugator)

* 과학기술처 연구비에 의한 다목적 공동설계 연구(MIC 86) 개발 결과임.

는 식(2)에 따라서 FFT연산과 IFFT연산 양식을 선택할 수 있게 한다.[5]

FFT연산 결과가 bit-reverse 순서로 얻어지므로, 출력 버퍼 메모리에서 Host프로세서로 전송될때, Host프로세서의 메모리 어드레스 계수기(address counter)을 bit-reverse함으로서, 올바른 순서로 load되게 한다.[4]

on-chip 어레이 제어회로는 데이터의 I/O 와 데이터 이동과정을 제어하며, 셀 내의 제어회로를 enable 시켜, IBA를 제어한다.

단 데이터 이동시와 버터플라이 연산시 필요한 클락주기 제약조건 차이를 고려하여, 2개의 클락으로 제어하는 기법을 사용하였다.[10]

즉 데이터 배열과 입 출력 과정에서는, 버터플라이 연산시보다 2배의 주파수를 갖는 클락을 사용하였다.

시스템 클락을 받아 그림(5)와 같은 2개의 클락을 발생시키는 clock generator가 칩내에 존재한다.

두개의 clock을 사용함에 따라 야기될 수 있는 플립플롭의 metastability현상을 방지하기 위해서, 동기화 회로(synchronizer)가 사용된다.

각각의 시스토크 셀의 구조는 그림(6)에 보여진다.

셀내에 TW look-up table 을 두개 씩에 따라 기존 연구에서[4] 필요 하였던 회절인자 loading과정이 없어져서, 버터플라이 입력 레지스터A,B는 데이터 이동 레지스터A,B와 각각 공유하게 되어 2개의 레지스터를 줄일수 있었다. 버터플라이 회로는 면적과 속도의 상반관계를 고려하여, Distributed 연산과 Merged 연산을 사용하는 2 adder-based구조로 되어 있다.[4]-[5]

데이터 이동클락은 그림(7)과 같은 구조를 가지며, 매stage에서 필요한 데이터 이동을 제어한다.[4]-[5]

셀 면적을 감소시키기 위해서, 버터플라이 회로에 사용되는 가산기클, BIC(binary lookahead carry)가산기에서 CP(carry propagation)가산기로 대체 시켰다.

CP가산기의 회로도 는 그림(8)과 같다.

설계된 칩의 테스트클 용이하게 하기 위해서 built-in self test하드웨어를 추가하여, test문제를 해결하였다.[11]

4. 시뮬레이션 및 결과 고찰

N=16인 경우, normal FFT동작 모드(TMS=0)와 test 모드(TMS=1)을 포함한 제어흐름도는 그림(9)와 같다.

여기서 TDIO는 test시 초기값의 load와 테스트결과를 output-buffer메모리로 출력하는 동작을 구별하는 제어신호이다.

시뮬레이션에 사용된 클락CLK2와 CLK는 각각25MHZ와 12.5MHZ이다.

시뮬레이션결과를 각각의 동작 모드에 따라 구별하여 나타낸 그림이 그림(10)과 같다.

위의 결과를 바탕으로 FFT연산시간은 약4400[ns]이며, 테스트시간은 약 21.4[us]가 소요된다.

5. 결론

본 논문은 집적도를 고려하여, N=16으로 고정된 FFT프로세서를 시스토크 야기텍처로 실현하는 연구이다. On-cell회절인자 look-up table을 활용으로서, 데이터 이동 레지스터와 버터플라이 입력 레지스터를 공유시킴으로써, 면적감소를 얻을 수 있었다.

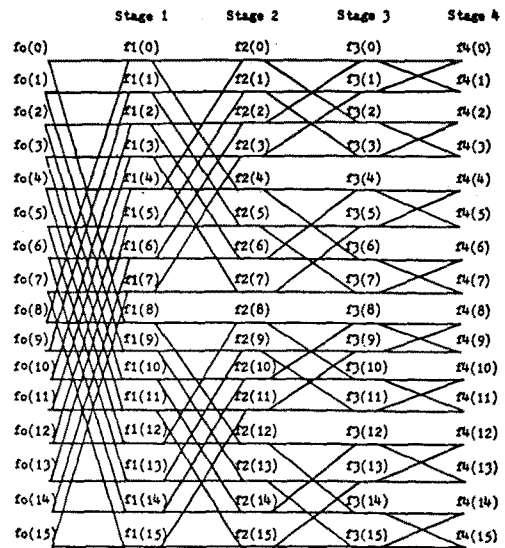
설계된 칩의 효율적인 테스트클 위해서 built-in self test회로를 추가하였는데, 이에 따라 야기되는 면적과 속도의 손실은 미소하다.

본 논문에서 설계된 프로세서는 규칙적인 하드웨어 구조를 갖고 있으므로, process기술이 발달하거나, WSI(wafer scale integration)기술이 실용화 되면 더 큰 N값의 FFT연산에도 적용이 가능하다. 앞으로의 연구방향은, 설계되고 있는 FFT프로세

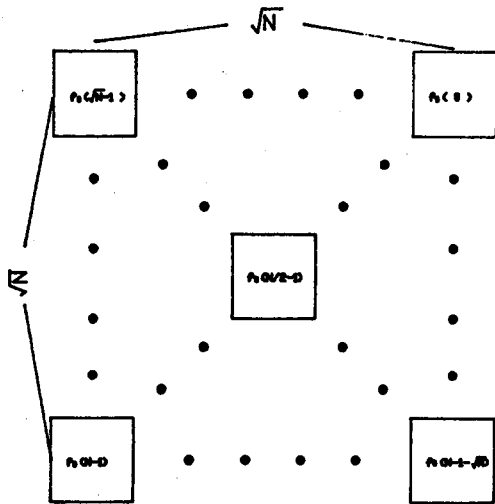
서를 IBM-AT와 interface시켜, 효율적인 응용에 중점을 두어야 한다.

6. 참고문헌

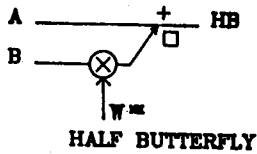
- (1) D.G.Fairbairn, "VLSI: A New frontier for system designers", Computer, Jan,1982 (pp,87-96)
- (2)C.D.Tompson, "Fourier transform in VLSI", IEEE Trans, Computer, Nov. 1983, (pp 1047-1057)
- (3)신경욱, "집적회로화된 FFT연산용 시스토크 어레이의 설계", 연세대학교 석사학위논문, 1985,12
- (4)최병운, "2차원 시스토크 FFT프로세서의 설계", 연세대학교 석사학위논문
- (5)이봉기, 신경욱, 최병운, 강병훈, 이장규, "다목적 공동설계(MPC)개발에 관한 연구 중간보고서", 과학기술저, 1987,2
- (6)G.L.Kratz and W.W.Sproul, "A microprogrammed approach to signal processing", IEEE. ASSP Aug,1974
- (7)H.T.Kung, "Why systolic architectures", computer, Jan,1982 (pp 36-46)
- (8)P.S.Liu and T.Y.Young, "VLSI Array Design under constraints of limited I/O bandwidth", IEEE Trans computer, Dec,1983 (pp 1160-1170)
- (9)M.A.Franklin, "Pin limitations of partitioning of VLSI interconnection networks", IEEE,Trans, computer. Nov,1982, (pp 1109-1116)
- (10)F.Ancenu, "A synchronous approach for clocking systems", IEEE, JSSC, Feb, 1982 (pp 51-56)
- (11)James Beausang and Alexander Albickli, "Incorporation of the BILBO technique within the existing chip design", IEEE 1985 Custom Integrated circuits conference(pp328-331)
- (12)Paul Wiley, "Interfacing Peripherals directly to an array processor", computer design, Aug,1979 (pp158)



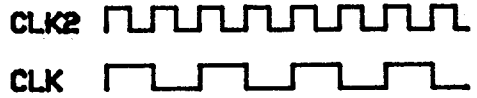
< 그림 1 > FFT연산 흐름도 (N=16)



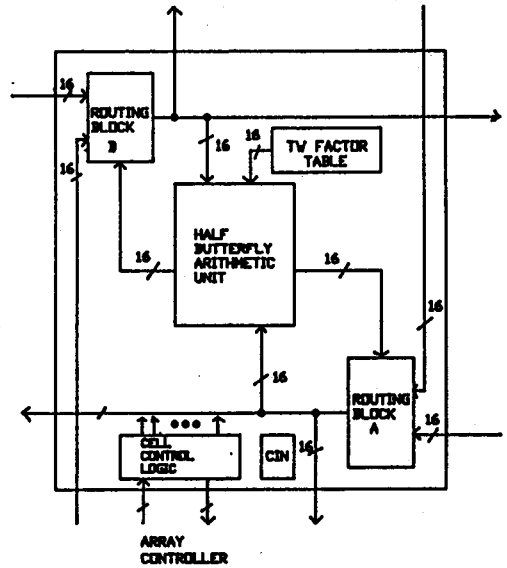
< 그림 2 > 2차원 시스토릭 어레이에 배열된 초기의 데이터값



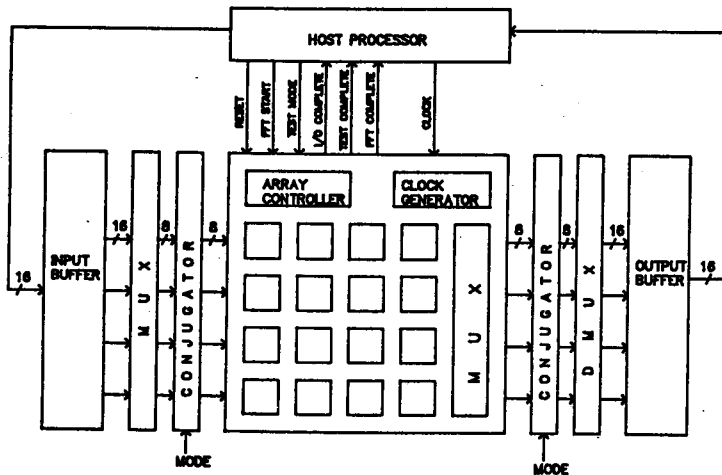
< 그림 3 > 반쪽 버터플라이 연산 (HBA: half butterfly arithmetic)



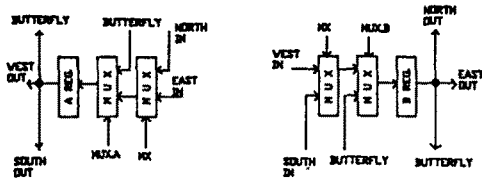
< 그림 5 > 칩에 사용된 2개의 클럭



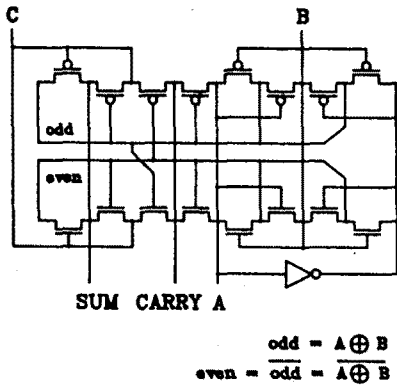
< 그림 6 > 셀의 구성도



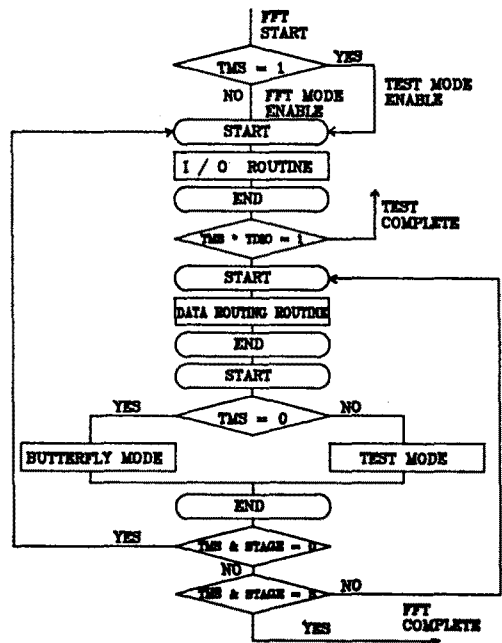
< 그림 4 > FFT프로세서의 구성도 (N=16)



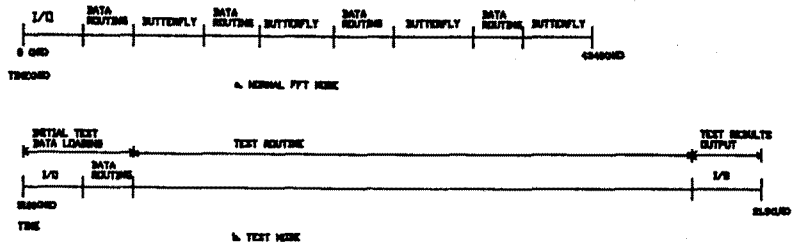
< 그림 7 > 데이터 이동 블록 A, B



< 그림 8 > Carry propagation 가산기의 1 bit full adder의 회로



< 그림 9 > 테스트 모드를 포함한 전체 FFT제어 흐름도 (N=16)



< 그림 10 > FFT프로세서에 대한 시뮬레이션 결과(N=16)