

FPGA 상에서 에너지 효율이 높은 병렬 신호처리 기법

장 주 옥[†] · 황 윤 일^{**} · Ronald Scrofano^{***} · Viktor K. Prasanna^{****}

요 약

본 논문은 알고리즘 레벨에서 FPGA를 이용하여 에너지 효율이 높은 기법을 제안한다. 제안한 기법을 기반으로 FFT와 행렬곱셈용 신호처리 응용을 위한 고효율 설계 기술을 제안한다. 또한 이러한 신호처리 응용 수행시 지연시간과 에너지 효율 측면에서의 FPGA 성능을 분석한다. Xilinx Virtex-II를 대상으로 Virtex-II Pro와 Texas Instrument TMS320C6415에 내장되는 Power PC 코어에서 구동되는 Xilinx library와 기존 알고리즘을 본 논문 기법과의 성능 비교를 수행한다. 성능 비교는 high-level에서 에너지와 지연 시간에 대한 유도 공식을 통한 추정치와 low-level 시뮬레이션을 통해 평가하였다. FFT에 대해 본 논문에서 제안한 기법은 Xilinx library와 DSP에 비해 각각 60%, 56% 적은 에너지를 소모한다는 결과를 얻었다. 또한 임베디드 프로세서와 비교해 EAT지수에서 10배의 개선을 보여준다. 위와 같은 결과는 FPGA가 DSP나 임베디드 프로세서에 비해 월등한 성능을 보여준다는 견해에 결정적인 단서가 된다. 또한, 이는 FPGA가 앞의 두 종류의 디바이스에 비해 더 적은 전력을 소모하면서 동시에 더 나은 성능을 보인다는 사실을 보여준다.

키워드 : 저전력 설계, FPGA, FFT, 매트릭스 곱셈기, 성능 예측

Energy-Efficient Signal Processing Using FPGAs

Ju-wook Jang[†] · Yunil Hwang^{**} · Ronald Scrofano^{***} · Viktor K. Prasanna^{****}

ABSTRACT

In this paper, we present algorithm-level techniques for energy-efficient design at the algorithm level using FPGAs. We then use these techniques to create energy-efficient designs for two signal processing kernel applications: fast Fourier transform(FFT) and matrix multiplication. We evaluate the performance, in terms of both latency and energy efficiency, of FPGAs in performing these tasks. Using a Xilinx Virtex-II as the target FPGA, we compare the performance of our designs to those from the Xilinx library as well as to conventional algorithms run on the PowerPC core embedded in the Virtex-II Pro and the Texas Instruments TMS320C6415. Our evaluations are done both through estimation based on energy and latency equations on high-level and through low-level simulation. For FFT, our designs dissipated an average of 60% less energy than the design from the Xilinx library and 56% less than the DSP. Our designs showed an EAT factor of 10 times improvement over the embedded processor. These results provide a concrete evidence to substantiate the idea that FPGAs can outperform DSPs and embedded processors in signal processing. Further, they show that FPGAs can achieve this performance while still dissipating less energy than the other two types of devices.

Key Words : Energy Efficient Design Techniques, FPGA, FFT, Matrix Multiplication, Performance Estimation

1. 서 론

FPGA는 강력한 프로세싱 능력과 주문제작 가능하다는 장점으로 인해 신호처리응용 구현에 많이 적용되고 있다. FPGA에 다수의 내장 곱셈기와 같은 기능추가는 이러한 장

점 중 하나이다. 현재 FPGA는 신호처리와 같은 계산을 요하는 응용에서의 적용 범위가 넓어지고 있다. 전통적으로 지연시간과 쓰루풋은 일반적인 프로세싱을 포함한 신호처리를 위한 성능 평가기준이 되어 왔다. 그렇지만, 휴대기기의 확산은 시스템의 연산처리능력뿐만 아니라 얼마나 에너지 효율적인가가 중요한 요소로 대두된다. 한 예로 소프트웨어-정의 라디오(SDR)[6]를 들 수 있다. SDR의 적응성과 높은 연산처리능력을 요함으로써 FPGA 기반 시스템의 성장 가능성이 높다.

현재 시장에는 상용 FPGA로 수천만 게이트가 내장된 저전력 성능을 가진 제품이 없다. 그러므로 본 논문에서는 FPGA에서의 저전력 최적화 기술 대신에 신호처리응용에서

* 본 논문은 정보통신부의 출연금으로 수행한 IT SoC 핵심설계인력양성사업의 수행결과임.

† 정 회 원 : 서강대학교 전자공학과 교수

** 준 회 원 : 서강대학교 전자공학과 석사과정

*** 비 회 원 : Dep. of Electrical Engineering University of Southern California, Ph.D Student

**** 비 회 원 : Dep. of Electrical Engineering University of Southern California, Professor

논문접수 : 2005년 3월 15일, 심사완료 : 2005년 6월 3일

의 에너지 소모를 최소화하는 알고리즘을 제안한다. 이는 저전력, 고속 연산처리 능력을 지닌 차세대 FPGA에 적용 가능하다. 본 연구에 적용된 기술은 FFT와 매트릭스 곱셈기의 신호처리 커널 응용을 위한 아키텍처와 알고리즘 설계에 적용될 수 있다. FFT는 SDR이나 센서 네트워크에서 이용되는 것과 같은 광대역 beamforming 응용에서의 compute-intensive portion이다. 행렬 곱산 역시 모바일과 SDR 시스템의 신호 영상 처리 시스템에서 자주 쓰이는 커널 연산이다. 본 논문에서는 같은 신호처리 커널 응용 환경에서 대표적인 DSP, 임베디드 프로세서인 Xilinx IPcore와 에너지 효율적 설계를 적용한 본 제안 방식간의 지연시간과 에너지 소모를 비교한다. 비교 실험은 지연시간과 에너지 방식을 토대로 High-level과 low-level에서의 모의실험을 통해 수행된다.

본 논문의 구성은 다음과 같다. 2절에서는 FPGA에서의 에너지 소모의 원인을 기술하고 이를 최소화 하는 알고리즘 기법들을 소개한다. 3절에서는 제안된 기법을 적용한 두 개 신호 처리 커널 응용을 기술한다.

2. 에너지 효율적 설계 기법

본 절에서는 FPGA 기반 설계에서 에너지 효율을 높이기 위한 기법들에 대해 다룬다. 전력과 에너지의 정의가 다음에 주의해 주기 바란다. 에너지는 평균 전력 소모와 지연시간의 곱으로 정의 된다. 에너지 소모를 이해하기 위해서는 전력 소모와 그에 따른 지연시간에 미치는 영향을 상호적으로 고려해야만 한다.

2.1 FPGA에서의 에너지 소모

FPGA 전력 소모에 관한 최근 연구는[13, 18, 19, 20]에서 언급된다. 이 연구들은 FPGA 기기에서의 전력소모가 내부연결에 주요인을 둔다고 제시한다. Xilinx Virtex-II[17]계열에서 전체 소모 전력중 50%~70%에 해당하는 에너지가 내부연결에서 소모된다고 보고한다. 나머지 전력소모는 clocking, 로직, 입출력 블록에서 일어난다. 이 분석은 클록 분배가 전력소모의 대부분을 차지한다는 ASIC 기술과는 차이점을 보여준다[18]. 이는 위 두 가지의 내부 연결 구조가 상이하므로 각기 다른 전력 소모 요인을 가지기 때문이다. FPGA의 내부연결 같은 경우는 미리 제작된 여러 길이의 전선 세그먼트와 라우팅 스위치로 이루어져 있다. 라우팅 스위치의 개선을 통한 전력 소모를 줄이기 위해 고속, 저전력, 대기상태를 두고 고속 동작은 기존의 방식을 따르고, 저전력 소모를 위해 속도를 줄이거나 대기상태로 전환하는 방법도 가능하다. 이를 위해 기존 배선 스위치를 개선해 기존 FPGA 내부연결과 연동이 가능하게 한다[19].

FPGA에서 또 다른 중요한 전력 소모 요인은 자원 소모 효율이다[13]. 일반적 FPGA 설계에서, 구성이 이루어지고 난 후 대부분의 자원은 이용되지 않으며 그에 따른 동적 전력 소모는 일어나지 않는다. 클록 주기에 서 신호변환 개수

에 따른 스위칭 활동 또한 전력 소모의 중요한 요인으로 지목된다. 각 자원에서의 스위칭 활동은 설계방법 뿐만 아니라 입력에 의한 영향을 받는다.

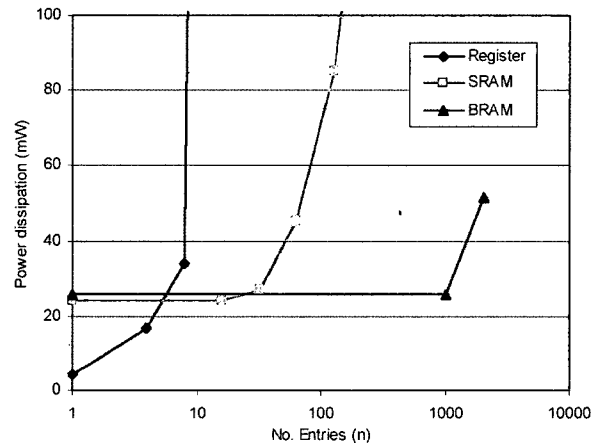
전력 소모 요인에 대한 이해를 바탕으로 low-level, 알고리즘 수준에서의 에너지 효율적 설계 기법에 대해 논의한다.

2.2 Low-level 설계 기법

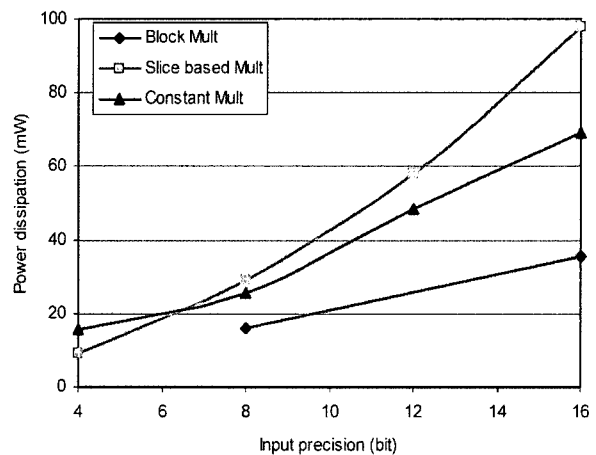
FPGA 설계에 응용시 에너지 절약을 위한 Low-level 전력 관리 기법은 여러 논문에서 제시되어 있다[13, 18]. 여기서는 알고리즘 수준 설계를 위한 조정에 필요한 low-level 기술들에 대해 다룬다.

이러한 기법의 하나로 클록 게이팅이 있다.

이는 기기에서 연산처리지 작동하지 않는 부분의 동작을 멈추게 하는 방법이다. Virtex-II 계열에서 클록 게이팅은 고주파수 클록에서 저주파수 클록으로 변환시키는 BUFMUX와 같은 프리미티브를 이용해 구현된다[17]. BUFMUX는 로직이 사용될 때에만 클록 트리를 동적으로 구동시킬 때 사용된다.



(그림 1) 다양한 저장 요인 구현시 엔트리의 개수와 전력 소모 간 상관관계(Virtex-II XC2V1500, 150MHz, 50% 변환 동작시, 입력당 16 비트)



(그림 2) 다양한 곱셈기에서 정확도와 전력 소모간 상관관계

다른 방법으로는 에너지 효율이 높은 바인딩을 선택하는 기법을 들 수 있다. 바인딩은 연산과 FPGA 컴포넌트를 사상하는 것을 의미한다. 적절한 바인딩을 선택해야 하는 이유는 같은 연산에 대한 여러 구성이 존재하기 때문이다. 그러므로 다른 바인딩들은 FPGA에서의 에너지 소모에 영향을 끼친다. 예를 들면, (그림 1)에서는 Virtex-II FPGA에서의 저장 방식으로 레지스터, SRAM과 BRAM 등의 세가지 방식을 비교한다. BRAM은 48개 엔트리 이상의 대용량 엔트리 필요시 다른 구현 방식에 비해 적은 전력 소모를 보인다. 다른 예로는 hard와 soft IP간의 선택을 들 수 있다. 이와 같은 경우에는 곱셈기 선택시 블록 곱셈기의 경우 Xilinx Virtex-II와 Altera Stratix가 CLB-based 곱셈기에 비해 월등한 효율을 보인다[(그림 2)참조]. (그림 1), (그림 2)의 모든 결과치는[3]에서 제안한 기법을 통해 얻어진다. 알고리즘 개발에 있어서 설계자는 설계 요구에 따른 다양한 바인딩에 의한 트레이드-오프를 분석할 수 있다.

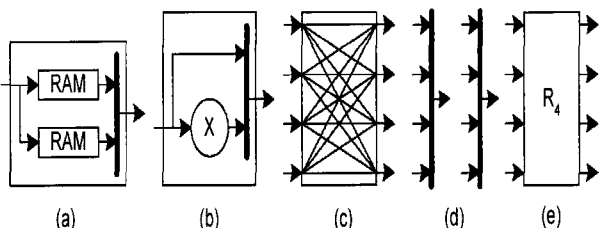
3. 신호처리 응용을 위한 에너지 효율적인 설계 기법

알고리즘 수준 기법을 적용해 두 가지의 응용을 FPGA에 매핑한다. FFT와 행렬 곱셈 응용은 신호 처리에서 매우 중요한 커널 응용이다. 이들 응용에 대해 이 장에서는 본 연구의 제안 방식과 앞서 언급한 설계 기법을 더불어 소개한다.

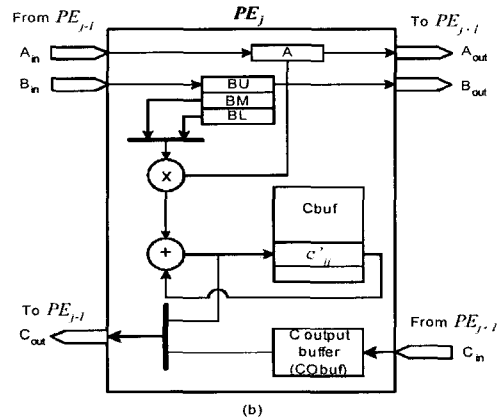
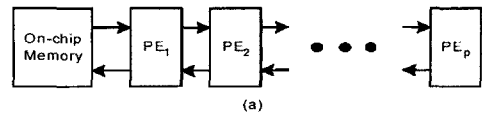
3.1 고속 푸리에 변환

본 연구에서는 FFT설계를 함에 있어, 잘 알려진 Cooley-Tukey 방법을 이용한다. N-point FFT 계산은 각 $\log_2(N)$ 단계마다 $O(N)$ 연산을 필요로 하므로, 총 $O(N\log_2N)$ 번의 연산을 수행해야 한다[10].

이러한 이유로 인해 FFT는 많은 양의 데이터를 처리하기 위해 파이프라인 구조를 쓴다. N-point FFT 설계는 radix-4 알고리즘을 기초로 한다. FFT 아키텍처에 쓰이는 매개변수들과 에너지 소모에 영향을 미치는 요인들의 상관관계를 통해, 설계방법을 선택한다. N-point FFT 설계에는 3가지 매개변수들이 중요하다: 1) 문제크기 (N), 2) 수평 병렬 정도 (H_p), 3) 수직 병렬 정도 (V_p). 수평 병렬 정도는 radix-4단계의 수가 병렬 ($1 \leq H_p \leq \log_4 N$)에서 얼마인가를



(그림 3) (a) 데이터 버퍼(Dbuf), (b) Twiddle 요인 계산(Twiddle), (c) 데이터 경로 순열, (d) 병렬-직렬/직렬-병렬 멀티플렉서, (e) Radix-4 계산 (R_4)



```

For t=1 to n do
For all j, 1 ≤ j ≤ n, do in parallel
PEj shifts data in BU right to PEj+1
If (BU=bkj), copy it into BM
For t=n+1 to n2+n do
For all j, 1 ≤ j ≤ n, do in parallel
PEj shifts data in A, BU right to PEj+1
If (BU=bkj), copy it into BL or BM (alternately)
If (A=aik), cij' := cij' + aik × bkj
(bkj is in either BM or BL, cij' is in Cbuf)
For t=n2+1 to 2n2 do
For all j, 1 ≤ j ≤ n, do in parallel
PEj store input Cm to Cobuf
for t=1 to n do
PEj output data cij' to PEj-1
for t=1 to n2-n do
PEj output data Cobuf to PEj-1
    
```

(그림 4) (a) 선행 배열 아키텍처, (b) 처리 소자, (c) 곱 행렬 알고리즘

정해준다. 수직 병렬 정도는 병렬에서 동작하는 입력의 개수를 정하는데 쓰인다. Radix-4 알고리즘을 통해 4개까지의 병렬 입력이 가능하다. 본 연구에서는 radix-4 butterfly, 데이터 버퍼, 데이터 경로 순열, 병렬-직렬/직렬-병렬 멀티플렉서, twiddle요인 연산처리 모듈 등의 다섯 개의 기본 모듈을 고려한다. 각각의 모듈은 매개변수화되고, 범용성을 가지므로, 기본모듈의 조합으로 완성된 설계가 가능하다[(그림 3) 참조].

3.2 행렬 곱셈

행렬 곱셈 알고리즘은 두 개의 $n \times n$ 입력 행렬 A, B 의 곱 $C = A \times B$ 로 이루어진다. 이 커널 응용을 위한 아키텍처는 [8]에서 제시되어 있다. [8]에서 제안된 아키텍처는 낮은 지연시간에 최적화 되어있다. 데이터 스트림 위에 행렬곱셈이 수행시, 쓰루풋은 중요한 성능 분석 기준이 된다. 본 제안 아키텍처는 쓰루풋-기반에 기초해 병렬처리와 파이프라인 방식을 적용한 것이다. 또한, [8]에는 없는 곱 행렬 C 산출시 필요한 논리를 포함하고 있다. 하나의 곱 행렬이 다음 연산과 겹쳐 수행되므로 산출시의 대기 주기가 없다. 이런 방식으로 한 클럭당 데이터 샘플의 쓰루풋이 얻어진다.

(그림 4)에서는 아키텍처와 알고리즘을 보여준다. 선형 배열 아키텍처가 적용되어 있다.

온-칩 메모리가 이용되었다. 이는 모든 행렬들이 Virtex-II 처럼 온-칩 메모리에 저장되어 있음을 보여준다. 적용된 알고리즘은 입력 행렬들을 선형 배열을 통해 빈틈없이 입력시켜 지연시간과 에너지 면에서 곱셈 연산을 효과적으로 수행한다. 행렬 A를 통한 입력 값이 선형 배열의 행에 입력되는 동안 행렬 B를 통한 입력 값들이 열에 입력된다. 여기서, 행렬 A를 통한 입력은 행렬 B에서의 입력이 끝날 n 주기 동안 선형배열에 입력되지 않는다.

PE들은 행렬 C에서 받은 곱셈 값들의 덧셈을 연산한다. 예로, (그림 4) (c)에서 n 값과 같은 개수의 PE들이 존재할 경우의 알고리즘을 나타낸다.

본 연구의 알고리즘에서, A, BU, BM과 BL은 PE안의 저장 레지스터들이다. a_{ik} 는 행렬 A의 i 번째 열, k 번째 행의 입력이다. b_{jk} 는 행렬 B의 k 번째 열, j 번째 행에 해당한다. PE_j 의 산출값은 PE_{j-1} 에 저장된다. (n^2+1) 번째 주기에서 시작해, n^2 의 결과값 c'_{ij} 는 PE에서 다음 $2n-1$ 주기동안 연산된다. 마지막 행렬 곱셈에서의 c'_{ij} 출력 값이 다음 주기의 중간출력값 c'_{ij} 에 의해 중복 쓰기 되는 것을 피하기 위해, 메모리(C0buf)는 각 PE에 존재한다. PE_{j-1} 은 C0buf의 cin 포트를 통해 출력값을 저장한다. n 주기 동안, PE_j 의 마지막 c'_{ij} 는 PE_{j-1} 에서의 출력값이다. 이후의 n^2-n 주기동안, C0buf안의 PE_j 에 저장된 c'_{ij} 는 PE_{j-1} 로 출력된다. PE_1 에서 출력되어 온-칩 메모리에 저장된 값들은 최종 결과값들이다. 각 클럭 주기마다 한 결과 값이 출력되므로 수행시간은 n^2 이다.

본 제안 선형 배열 설계는 이웃하는 PE간의 연결을 보장하고 내부연결을 최단 거리로 구성한다. Virtex-II에 구현된 본 제안 방식의 설계에서 내부연결은 거의 직접 혹은 이중 연결로 이루어진다. 행렬 A와 B의 모든 데이터는 파이프라인 방식으로 왼쪽에서 오른쪽으로 입력된다. 각각 PE들은 언제나 작동하므로 쓰루풋이 증가된다. 에너지 효율적인 바인딩은 곱셈기, CBuf와 C0buf에 적용된다. 입력단의 개수가 3보다 커진다면, 에너지 효율 측면에서 SRAM이 이용될 것이다. 블록 곱셈기는 특히 두 개의 입력이 상수값이 아닐때 에너지 효율이 높다. 추가로 이용된 설계 기법은 블록 행렬 곱셈으로 $n < 24$ 일때 n 개의 PE를 사용하는 완전 병렬 아키텍처를 적용한 경우가 좋은 예이다. $n > 24$ 일때에는 블록 크기 (n/p) 를 이용한 블록 행렬 곱셈을 이용한다. 여기서 p 는 PE의 개수이다. 이 기술은 쓰루풋을 감소시키는 반면에 면적과 에너지를 절약한다.

본 제안 알고리즘은 두 개의 $n \times n$ 행렬을 $(n/p)^3 (2p^2)$ 주기동안 연산한다. 파이프라인방식은 효율적 지연시간을 절반으로 감소시킨다.

4. Low-level 모의실험 결과

4절에서 high-level 에너지 소모의 정확도를 실험하기 위해, 종합된 설계와 low-level 모의실험을 바탕으로 한 실제 값들과 방정식을 통한 평가를 비교하였다. Low-level 모의 실험은 100과 150MHz에서 구동되는 Virtex-II FPGA, 500MHz에서 구동되는 TMS320C6415 DSP, 300MHz에서 구동되는 PowerPC 등을 이용한 FFT와 행렬 곱셈을 통해 수행된다. 본 연구는 제안 설계를 같은 FPGA상에서 비교하기 위해 Xilinx 자료집을 토대로 FFT와 행렬 곱셈을 종합하였다.

Virtex-II에 대한 결과 값들은 본 제안 설계를 적용한 VHDL 코딩을 이용한 low-level 모의실험에 기초하여 얻어진다. 이들 제안 설계는 Xilinx ISE 4.1i 내장 XST(Xilinx synthesis Technology)를 이용해 종합되었다. 배치와 배선 관련 파일(.ncd file)은 Virtex-II XC2V1500(package bg575, speed grade -5)와 XC2V3000(package bg728, speed grade -5)를 통해 얻어진다. 모의실험을 위한 입력 시험 벡터들은 랜덤 발생되고 평균 스위칭 활동은 50%이다. 본 설계의 모의 실험과 결과 값들(.vcd file)은 Mentor Graphics ModelSim 5.5e를 통해 얻어졌다. 이 두 파일들은 다시 평균 전력 소모를 구하기 위해 Xilinx XPower에 사용된다. 에너지 소모는 평균 전력이 지연시간을 곱하여 얻어진다. 에너지 소모에 대해 본 제안 방정식을 이용한 평가 오차는 20% 이하로 나타났다. 또한, 정확도를 기하기 위해 통계 분석을 이용하였다. 표본 중간값 에너지 소모에 대한 신뢰 간격을 이용한다. 오차범위는 입력 값에 대한 신뢰도를 높인다. 이는 오차범위가 전체량에 대한 중간값이 실험을 통한 중간값의 정적 범위 안에 있기 때문이다. \bar{x} 가 실험을 통해 얻어진 샘플 중간값, α 는 0과 1사이의 값, $z_{\alpha/2}$ 는 상수[7], M 은 샘플의 개수를 나타낼때, 방정식 $\bar{x} \pm z_{\alpha/2} (s/\sqrt{M})$ 을 통해 모의실험에 대한 오차범위를 평가한다. 행렬 곱셈에 대한 에너지 소모를 통계적으로 분석하기 위해, 50개의 다른 $n \times n$ 행렬 곱셈을 수행하였다. 위와 같은 Low-level 모의실험 과정동안 고르게 분산되고 랜덤 발생 행렬들을 입력으로 사용한다. 예로, $n=15$ 일때, $1509.72nJ$ 의 평균 에너지 소모가 ± 1.85 오차 범위내에서 95%의 신뢰도를 갖는다.

<표 2>와 <표 4>는 본 제안 설계와 Xilinx 방식, DSP 방식간의 성능 비교를 나타낸다[16]. Xilinx는 CoreGen 자료집 내에서 다양한 크기의 FFT를 제공한다. 여기서 Xilinx

<표 1> 행렬 곱셈에 대한 에너지 소모

Problem size (n)	Logic E (nJ)	Interconnect E (nJ)
3	8.99	6.99
6	58.82	47.09
12	421.73	353.24
15	785.55	724.17

〈표 2〉 Xilinx 자료집 기반 설계, TI DSP, 본 제안 설계의 FFT 성능

Problem size(n)	Xilinx(100MHz)				TI DSP(500MHz)				Our Designs(100MHz)								
	A	T	Em	EAT	T	Eest	Em	E err	Vp	Hp	Binding	A	T	Eest	Em	E err	EAT
16	136	0.16	179.6	0.04	0.17	183.2	199.9	9%	1	2	SRAM	1171	0.16	65.4	77.0	15%	0.014
									4	2	SRAM	2390	0.04	63.5	75.2	15%	0.007
64	1079	1.92	1785.6	3.70	0.60	656.8	716.4	9%	1	3	SRAM	2266	0.64	552.4	493.3	12%	0.72
									1	3	BRAM	1613	0.64	464.2	390.4	19%	0.40
									4	3	SRAM	5690	0.16	393.9	418.7	6%	0.38
									4	3	BRAM	4193	0.16	403.2	400.4	1%	0.27
256	1303	7.68	6927.3	69.32	2.53	2958.3	3008.3	2%	1	4	BRAM	2050	2.56	2582.2	2223.1	16%	11.67
									4	4	BRAM	5624	0.64	2203.2	1971.3	12%	7.10
1024	1557	30.72	34283.5	1639.82	12.07	14284.7	14365.7	1%	1	5	BRAM	2744	10.24	14963.5	13739.4	9%	386.06
									4	5	BRAM	6673	2.56	11424.7	9204.2	20%	157.23

Eest is the estimated energy (nJ). Em is the measured energy (nJ) from the synthesized designs.
The unit of EAT is 1E-12. The unit of Area (A) is slicio, The unit of time (T) is usec.

〈표 3〉 Xilinx 자료집 기반 설계와 TI DSP 사이의 FFT 성능 비교

Problem size(n)	Our designs(100MHz)			Our designs vs. Xilinx				Our Designs vs. DSP	
	Vp	Hp	Binding	E(decrease)	A(increase)	T(decrease)	EAT(decrease)	E(decrease)	T(decrease)
16	1	2	SRAM	57%	0.86x	1.0x	2.71x	61%	1.06x
	4	2	SRAM	58%	1.75x	4.0x	5.45x	62%	4.25x
64	1	3	SRAM	72%	2.10x	3.0x	5.17x	31%	0.94x
	1	3	BRAM	78%	1.49x	3.0x	9.18x	46%	0.94x
	4	3	SRAM	77%	5.27x	12.0x	9.70x	42%	3.75x
	4	3	BRAM	78%	3.89x	12.0x	13.77x	44%	3.75x
256	1	4	BRAM	68%	1.57x	3.0x	5.94x	68%	0.99x
	4	4	BRAM	72%	4.32x	12.0x	9.77x	72%	3.95x
1024	1	5	BRAM	60%	1.76x	3.0x	4.25x	60%	1.18x
	4	5	BRAM	73%	4.29x	12.0x	10.43x	73%	4.71x

〈표 4〉 행렬 곱셈에서의 에너지 소모

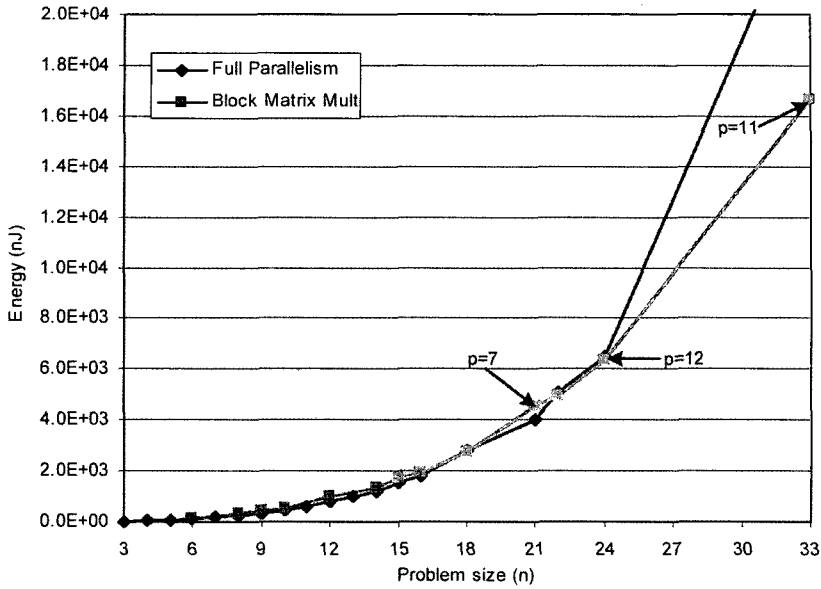
Problem size(n)	Xilinx(100MHz)				TI DSP(500MHz)				Our Designs(100MHz)					
	A	T	Em	EAT	T	Eest	Em	E err	A	T	Eest	Em	E err	EAT
3	299	0.18	23.12	1.24	0.14	149.94	166.6	11%	434	0.06	13.37	15.98	16%	0.42
6	299	1.44	184.98	79.65	0.38	454.58	456.96	1%	861	0.24	98.95	105.91	7%	21.89
12	299	11.52	1479.84	5097.29	1.54	1825.46	1827.84	0%	1699	0.96	755.95	774.97	2%	1264.00
15	299	22.50	2890.32	19444.63	3.22	3834.18	3836.56	0%	2083	1.50	1463.47	1509.72	3%	4717.12

Eest is the estimated energy (nJ). Em is the measured energy (nJ) from the synthesized designs.
The unit of EAT is 1E-12. The unit of Area (A) is slicio, The unit of time (T) is usec.

〈표 5〉 PowerPC(300MHz)에서 FFT와 행렬곱셈시의 전력 소모

FFT(300MHz)					Matrix Multiplication(300MHz)				
Problem size(n)	Tl(usec)	Tm(usec)	E/(nJ)	Em(nJ)	Problem size(n)	Tl(usec)	Tm(usec)	E/(nJ)	Em(nJ)
16	2.24	59.40	605	16038	3	0.09	34.98	24	9444
64	23.68	353.10	6394	95337	6	0.72	310.29	194	83778
256	183.47	n/a	49536	n/a	12	5.76	2311.35	1555	624065
1024	1206.40	n/a	325728	n/a	15	11.25	4436.64	3038	1197893

Tl is a lower bound of time (T). Tm is the measured time (T). El is a lower bound of energy (E). Em is the measured energy (E).



(그림 5) 완전 병렬 처리 아키텍처와 블록 행렬 곱셈시의 에너지 소모(150MHz, 데이터당 8 비트)

FFT는 100MHz 속도로 동작한다. 또한, Xilinx는 3×3행렬 곱셈만을 위한 최적화된 설계를 제공한다. 이 행렬 곱셈 설계는 150MHz에서 동작한다. $n > 3$ 일때, 블록 행렬 곱셈을 이용한다. TMS320C6415에 대한 결과값은 Texas Instruments의 Code Composer Studio 2.1을 통해 얻어진다. FFT와 행렬 곱셈에서, TI의 DSP 자료집내의 *DSP_fft*, *DSP_mat_mult* 알고리즘을 사용한다[16]. 지연시간은 오직 프로파일링을 통해 얻어진다. 에너지 소모를 계산하기 위해, 앞서 언급한 75% high/25% low 상태를 가정한다. <표 2>에도 나타나듯이, high-level 모의실험을 통한 에너지 소모 평가 오차는 11%이하로 보여진다.

이들 비교는 각 행렬들과 에너지, 변적 지연시간 간 곱의 행렬들(EAT)을 통해 얻어진다[2]. EAT 행렬을 이용해 여러 설계를 비교 할때, 가장 작은 EAT 값이 최적값이다. <표 3>는 Xilinx와 TI DSP에 대한 본 제안 설계의 성능 증가를 나타낸다.

FFT에서의 에너지 효율은 파이프라인 방식, 사용하지 않는 모듈의 기능억제와 효과적인 메모리 바인딩 선택을 통해 얻어진다. twiddle 요인 연산 모듈의 기능 억제를 통해 30%이상의 에너지 절약효과를 준다. 또한, 4진 기수 알고리즘을 통해 복소수 곱셈의 횟수를 줄인다. 바인딩 선택을 통해서는 에너지 효율을 높일 수 있다. 추가로, 병렬 처리 방식이 쓰루풋과 에너지 효율을 높이는 반면, 내부연결에서 사용하는 에너지양이 급격히 증가한다는 점을 확인할 수 있다. 예로, $N=256$ 일때 $(V_p, H_p)=(4, 1)$ 에서 $(V_p, H_p)=(1, 4)$ 보다 두 배 높은 쓰루풋을 보이는 반면, 약 20% 가량 많은 에너지를 소모한다. 더 많은 내부연결과 메모리 사용이 앞의 설계에서 에너지를 더 소모하는 이유가 된다.

행렬 곱셈에서의 에너지 효율은 지연시간 감소, 파이프라인 방식, 병렬처리방식과 에너지 효율적인 바인딩을 통해 확

득되었다. <표 1>는 로직과 내부연결에 대한 에너지 소모를 보여준다(XPower는 입출력을 포함해 이 정보를 제공해 준다). 일반적인 설계의 에너지 소모는 내부연결에서 대부분 발생하나[13], 본 제안 설계에서는 로직이 에너지 소모의 중요한 변수이다. 이는 선형 배열의 아키텍처 선택에 따른 것이다. (그림 5)은 선형 배열 아키텍처에서 행렬 곱셈시의 에너지 소모를 나타낸다. Problem size가 24보다 증가시, 블록 행렬 곱셈이 보다 효율적인 에너지 성능을 보여준다. 큰 problem size에 필요한 내장 저장장치의 크기가 급격히 증가함에 따라 병렬처리 방식은 총 에너지 소모량을 늘리는 단점을 지닌다. 이 결과는 설계자가 알고리즘과 병렬처리 정도 사이의 트레이드-오프를 충분히 인식하고 있어야 한다는 사실을 보여준다.

Virtex-II에서 구동되는 PowerPC 코어를 위해, 본 연구에서는 C 언어로 FFT와 행렬 곱셈 프로그램을 작성하였다. 이를 PowerPC 기반 gcc 컴파일러로 컴파일 하고, Xilinx Virtex-II 전문 개발자 키트를 이용한 모의실험을 수행하였다. PowerPC의 SWIFT 모델 모의실험을 실시하였다. SWIFT 모델 모의실험은 실제 PPC405 코드의 실행을 허용한다. 여기서 계산된 데이터는 Virtex-II안의 BRAM에 저장된다. 지연시간의 모의실험결과가 직접 나오는 동안, 에너지 결과값은 전력 소모 0.9mW/MHz, 클럭 주파수 300MHz라 가정하고 계산된다. 결과는 4장에서 얻어진 lower bound와 함께 <표 3>에 나타난다. PowerPC 설계에서의 상수 요인이 상대적으로 높은 이유는 모든 종류의 명령어중 산술 명령어만을 토대로 평가했기 때문이다. problem size가 증가하고 산술 명령어들이 우위를 점하기 시작하면, 비록 느리긴 하지만 상수 요인은 점차 감소한다. BRAM에서의 에너지 소모는 고려되지 않았음을 밝힌다. 이는 에너지 소모를 늘릴 것이다. 추후과제는 PowerPC에서 high-level 평가

기술의 정확도를 높이는 것이 목표이다.

6. 결론 및 추후과제

저전력, 고속 연산처리가 가능한 차세대 FPGA 설계에 적용가능한 신호처리 응용에서의 에너지 소모를 최소화하는 알고리즘을 제시하였다. 대표적인 DSP, 임베디드 프로세서인 Xilinx IPcore와 본 제안 방식간의 지연시간과 에너지 소모를 비교해 보았다. 모의실험은 지연시간과 에너지 방식을 토대로 Low-level, High-level에서의 모의실험이 포함되었다. FFT에 대해 본 논문에서 제안한 기법은 Xilinx library와 DSP에 비해 각각 60%, 56% 적은 에너지를 소모한다는 결과를 얻었다. 또한 임베디드 프로세서와 비교해 10배의 개선을 보여준다.

추후과제의 범위는 매우 넓다. 예로, DSP와 임베디드 프로세서에서의 에너지 소모량 예측의 정확도를 높이는 기술을 개발하는 것이다. 특히 관심이 높은 분야는 충격계수의 영향을 분석해 기기의 종류에 관한 것이다. 예로, 어떤 기기는 주어진 커널 응용에서 더 에너지 효율적인 반면, 유휴시간동안 에너지 소모가 높아지는 단점이 존재한다. 이런 경우에서 적절한 기기를 선택하는 것은 본 논문에서 서술된 분석의 종류만이 아니라, 각기 다른 기기에서의 저전력 모드의 유용성과 특정한 시나리오에 따른 사용에서의 분석을 포함한다.

참고 문헌

- [1] Altera Corp., <http://www.altera.com>. Apex20K data sheet, 2002.
- [2] B. Bass, "A Low-Power, High-Performance, 1024-Point FFT Processor," IEEE Journal of Solid-State Circuits, Vol. 34, No.3(1999), 380-38.
- [3] S. Choi, J.-W. Jang, S. Mohanty, and V. K. Prasanna, "Domain-Specific Modeling for Rapid System-Wide Energy Estimation of Reconfigurable Architectures," Engineering of Reconfigurable Systems and Algorithms, 2002.
- [4] E. Chu and A. George, Inside the FFT Black Box, CRC Press, 2000.
- [5] J. A. B. Fortes, K. S. Fu, and B. Wah, "Systematic Approaches to the Design of Algorithmically Specified Systolic Arrays," International Conference on Acoustics, Signal, and Speech Processing, 1985.
- [6] C. Dick, "The Platform FPGA: Enabling the Software Radio," Software Defined Radio Technical Conference and Product Exposition(SDR), November, 2002.
- [7] R. Hogg and E. Tanis, Probability and Statistical Inference, 6th Eds., Prentice Hall, pp.656-657, 2001.
- [8] J.-w. Jang, S. Choi, and V. K. Prasanna, "Energy Efficient

Matrix Multiplication on FPGAs," Field-Programmable Logic and Applications, 2002.

- [9] S. Lei and K. Yao, "Efficient Systolic Array Implementations of Digital Filtering," IEEE International Symposium on Circuits and Systems, 1989.
- [10] A. V. Oppenheim and R. W. Schaffer, Discrete-Time Signal Processing, Prentice Hall, 1989.
- [11] A. Raghunathan, N. K. Jha, and S. Dey, High-level Power Analysis and Optimization, Kluwer Academic Publishers, 1998.
- [12] R. Scrofano, S. Choi, and V. K. Prasanna, "Energy Efficiency of FPGAs and Programmable Processors for Matrix Multiplication," IEEE International Conference on Field-Programmable Technology, 2002.
- [13] L. Shang, A. Kaviani, and K. Bathala, "Dynamic Power Consumption in Virtex-II FPGA Family," International Symposium on Field Programmable Gate Arrays, 2002.
- [14] H. Styles and W. Luk, "Customising Graphics Application: Techniques and Programming Interface," IEEE Symposium on Field Programmable Custom Computing Machines, 2000.
- [15] R. Tessier and W. Burlison, "Reconfigurable Computing and Digital Signal Processing: A Survey," Journal of VLSI Signal Processing, May/June, 2001.
- [16] Texas Instruments, <http://www.ti.com>.
- [17] Xilinx Incorp., <http://www.xilinx.com>.
- [18] G. Yeap, Practical Low Power Digital VLSI Design, Kluwer Academic Publishers, 1998.
- [19] J. Anderson, F. Najm, "Low-power programmable routing circuitry for FPGAs," IEEE/ACM Conference on Computer Aided Design ICCAD-2004, pp.602-609.
- [20] J. Becker, M. Huebner, M. Ullmann, "Power estimation and power measurement of Xilinx Virtex FPGAs: trade-offs and limitations," Integrated Circuits and Systems Design, 2003.



장 주 욱

1983년 서울대학교 전자공학과(학사)

1985년 한국과학기술원(석사)

1993년 미국 University of Southern California 컴퓨터공학과(박사)

1985년~1994년 삼성전자 컴퓨터개발실

1995년~현재 서강대학교 전자공학과 교수

관심분야: 병렬처리, IT SoC, 인터넷 프로토콜



황 윤 일

2004년 서강대학교 전자공학과(학사)
2004년~현재 서강대학교 전자공학과
석사과정
관심분야: IT SoC, 인터넷 프로토콜



Viktor K. Prasanna

인도 Bangalore University 전자공학
(학사)
인도 Indian Institute of Science 전자공
학(석사)
미국 Pennsylvania State University 컴퓨
터학(박사)

관심분야: 연산처리, 병렬 분산 시스템, 네트워크 컴퓨팅, 임베
디드 시스템

Ronald Scrofano

현 재 미국 University of Southern California 전자공학 박사과정
관심분야: 슈퍼컴퓨팅, 병렬 분산 시스템, 네트워크 컴퓨팅