

고성능 멀티미디어 처리용 병렬프로세서 하드웨어 설계 및 구현

김 용 민*, 황 철 희**, 김 철 홍***, 김 중 면****

Hardware Design and Implementation of a Parallel Processor for High-Performance Multimedia Processing

Yong-Min Kim*, Chulhee Hwang**, Cheol-Hong Kim***, Jong-Myon Kim****

요 약

최근 모바일 멀티미디어 기기들의 사용이 증가 하면서 고성능 멀티미디어 프로세서에 대한 필요성이 증가하고 있다. 본 논문에서는 낮은 소비전력으로 고성능 멀티미디어 애플리케이션을 구현할 수 있는 SIMD기반 병렬프로세서를 제안한다. 제안하는 병렬프로세서는 16개의 프로세싱 엘리먼트로 구성되어 있으며, 3단계 파이프라인 구조로 설계되었다. 모의실험 결과, 제안한 SIMD기반 병렬프로세서는 기존의 병렬프로세서보다 프로세싱 엘리먼트 당 상대 연산 처리량에서 높은 성능을 보였으며, 또한 동일한 130nm 테크놀로지와 720 클록주파수에서 상용 고성능 프로세서인 TI C6416보다 1.4~31.4배의 성능 향상 및 5.9~8.1배의 에너지 효율 향상을 보였다. 제안한 병렬프로세서를 하드웨어 설계언어인 verilog HDL을 이용하여 설계하였고, FPGA를 이용해 검증하였다.

▶ Keyword : 병렬프로세서, 멀티미디어 처리, 파이프라인 아키텍처, 하드웨어 설계

Abstract

As the use of mobile multimedia devices is increasing in the recent year, the needs for high-performance multimedia processors are increasing. In this regard, we propose a SIMD (Single Instruction Multiple Data) based parallel processor that supports high-performance multimedia applications with low energy consumption. The proposed parallel processor consists of 16 processing elements (PEs) and operates on a 3-stage pipelining. Experimental results indicated

-
- 제1저자 : 김용민 • 교신저자 : 김중면
 - 투고일 : 2010. 11. 04, 심사일 : 2010. 12. 23, 게재확정일 : 2011. 02. 09.
 - * 울산대학교 전기공학부 (School of Electrical Engineering, University of Ulsan) 석사과정
 - ** 울산대학교 전기공학부 (School of Electrical Engineering, University of Ulsan) 석사과정
 - *** 전남대학교 전자컴퓨터공학부 (Department of Electronics and Computer Engineering, Chonnam National University) 교수
 - **** 울산대학교 전기공학부 (School of Electrical Engineering, University of Ulsan) 교수
 - ※ 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2011-0017941). 또한, 본 연구는 중소기업청에서 지원하는 2010년도 산학연공동기술개발사업(No. 000406420110)의 연구수행으로 인한 결과물임을 밝힙니다. 톨은 반도체설계교육센터(IDECE)의 지원을 받아 수행되었음.

that the proposed parallel processor outperforms conventional parallel processors in terms of performance. In addition, our proposed parallel processor outperforms commercial high-performance TI C6416 DSP in terms of performance (1.4-31.4x better) and energy efficiency (5.9-8.1x better) with same 130nm technology and 720 clock frequency. The proposed parallel processor was developed with verilog HDL and verified with a FPGA prototype system.

▶ Keyword : Parallel processor, multimedia processing, pipeline architecture, hardware design

1. 서론

최근 모바일 멀티미디어 기기들의 사용이 증가하면서 고성능 멀티미디어 프로세서에 대한 필요성이 증가하고 있으며, 프로세서는 더 이상 동작 주파수를 높이는 방법이 아닌 다수의 프로세서를 집적하는 멀티프로세서로 기술 발전이 이루어지고 있다 [1].

주문형반도체인 ASIC (Application-Specific Integrated Circuit)은 멀티미디어 애플리케이션에서 요구되는 고성능, 저비용 및 저전력을 만족시킬 수 있지만 다양한 애플리케이션에서 요구되는 범용성 혹은 유연성 (flexibility)을 만족시키지 못한다. 반면에 그림 1과 같이 DSP (Digital Signal Processor)나 범용 프로세서(General-Purpose Processor, GPP) 기반의 시스템은 범용성에 기인하여 다양한 응용 분야에 사용될 수 있으나 멀티미디어 애플리케이션에서 요구되는 높은 레벨의 성능을 만족시키지 못한다. 왜냐하면 DSP나 GPP는 프로세서 구조의 특성상 멀티미디어에 내재한 높은 병렬성 (massive parallelism)을 활용하지 못하기 때문이다 [2].

멀티미디어 애플리케이션을 위해 주문형반도체의 저비용, 저전력, 고성능과 DSP나 범용 마이크로프로세서의 유연성이 결합된 새로운 형태의 프로세서인 ASIP (Application Specific Instruction Set Processor)가 대안으로 부각되고 있다 [3]. 본 논문에서는 ASIP 모델 중 2차원인 이미지나 비디오 처리에 특화된 SIMD (single instruction multiple data)기반 병렬프로세서를 제안한다. 명령어 레벨 (instruction-level)이나 스레드 레벨 (thread-level) 프로세서들은 실리콘 면적을 멀티포트 레지스터 파일 (multiported register file), 캐쉬 (cache), 파이프라인 (deep pipelined) 기능 유닛 등으로 사용하는 반면, SIMD기반 병렬프로세서는 여러 개의 저비용 프로세싱 엘리먼트 (processing element, PE)들을 이용하여 고성능을 추구하고 동시에 저장장소와 데이터 통신 요구를 최소화하기 위해 프로세싱 엘리먼트와 데이터 입출력을 동일위치에 배치함으로써 저 전력을 만족시킨다. 또한 각 PE간의 상호 연결망 (interconnection network)은 짧은 내부연결선을 이용한 격자(mesh) 구조로 연결되어 있어 저 비용을 만족하는 동시에 지역성이나 규칙성 있는 2차원 패턴 이미지나 동영상 처리 [4]에 높은 병렬성을 보여 준다.

본 논문에서 제안한 SIMD기반 병렬프로세서는 16개의 PE로 구성되어 있으며, 3단계 파이프라인 구조로 설계되었다. 모의실험 결과, 제안한 병렬프로세서는 기존의 병렬프로세서보다 높은 성능을 보였으며, 상용 고성능 DSP보다 상당한 성능 (1.4~31.4배) 및 에너지 효율 (5.9~8.1배) 향상을 보였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 소개하고, 3장에서는 제안하는 병렬프로세서 구조를 소개한다. 4장과 5장에서는 제안하는 병렬프로세서의 기능적 특징 설명과 성능을 분석하고, 6장에서는 하드웨어 합성 결과를 보여준다. 끝으로 7장에서는 본 논문의 결론을 맺는다.

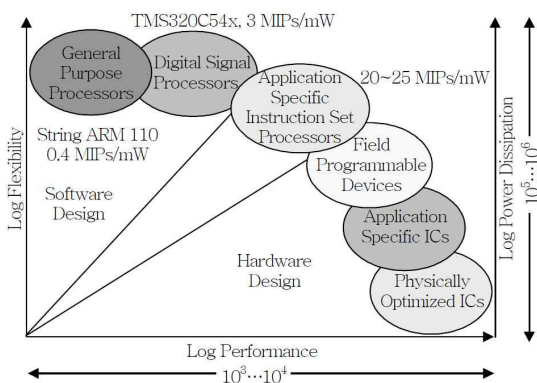


그림 1. 유연성 vs. 성능 (자료: CoWare Inc.)
Fig. 1. Flexibility versus performance

II. 관련 연구

멀티미디어 애플리케이션에 대한 데이터 레벨 병렬성 (data-level parallelism, DLP)에 관한 연구는 크게 두 개의 연구 그룹으로 나누어진다: (1) 현재의 SIMD 명령어를 이용하여 성능을 향상시키는 그룹 [5]-[9]과 (2) SIMD기반 병렬 프로세서를 이용하여 성능을 향상시키는 그룹 [10],[11]. 많은 연구 그룹 혹은 개인들이 범용 마이크로프로세서에서 멀티미디어 애플리케이션에 대한 SIMD 명령어의 효율성에 대하여 분석하였다. [5]에서는 UltraSPARC 프로세서에서 이미지와 비디오 처리에 대한 VIS 명령어의 효율성을 기술하였다. 4-way out-of-order 프로세서는 single in-order 프로세서보다 2.3배~4.2배의 성능을 향상시켰고 더불어 VIS (Visual Instruction Set) 명령어는 1.1 배~4.2배의 성능을 더 향상시켰다. [6]에서는 DSP와 멀티미디어 애플리케이션에 대한 MMX (Multi-Media eXtension) 명령어의 성능 평가를 기술하였다. MMX 명령어는 81%의 다이내믹 명령어를 감소시켜 평균 5.5배의 성능 향상을 보였다. 이러한 결과에서 보는 바와 같이 SIMD 명령어는 적당한 수준의 성능을 향상시킨다. 하지만 멀티미디어 애플리케이션에 내재한 완전한 데이터 병렬성을 얻지 못하기 때문에 다양한 형태의 멀티미디어에서 요구되는 상당한 양의

성능 요구를 만족시키지 못할 것이다.

SIMD기반 병렬 프로세서는 공간적 병렬성(spatial parallelism)을 실현하기 위해 여러 개의 동기화된 프로세싱 유닛(processing unit)들을 사용한다. 이 유닛들은 하나의 제어 유닛으로부터 동시에 전송되는 동일한 연산 명령어를 서로 다른 데이터에 대하여 수행한다. 따라서 데이터 병렬 모델을 이용하여 성능을 향상시킨다. 고도 데이터 병렬 어레이 (massively data parallel array)들은 거의 30년 동안 이미지 처리에 사용되어 왔지만, 초기의 SIMD기반 병렬 프로세서 (TMC Connection Machine 1 [12])는 I/O 테크놀로지에 의해 제한되었다. 이후의 SIMD 병렬 프로세서인 TMC CM-2 [13]와 MasPar MP-2 [14]는 버퍼 이미지의 큰 병렬 디스크 어레이의 사용을 통해 이러한 제한을 극복하였지만 큰 비용과 휴대성에서 문제가 있다. Fine-grained 병렬 프로세서인 MGAP [15]와 ABACUS [16]는 이러한 휴대성 이슈를 해결하였지만, 그들의 성능은 I/O bandwidth와 latency에 의해 제한되었다.

이러한 기존의 병렬 프로세서와 다르게, 본 논문에서 제안한 SIMD기반 병렬 프로세서는 프로세서와 센서의 직접적 연결을 통해 I/O 대역의 문제를 해결하고, 또한 짧은 와이어의 사용으로 높은 면적과 에너지 효율을 보이는 동시에 많은 데이터에 동일한 명령어를 수행하여 고성능을 추구한다.

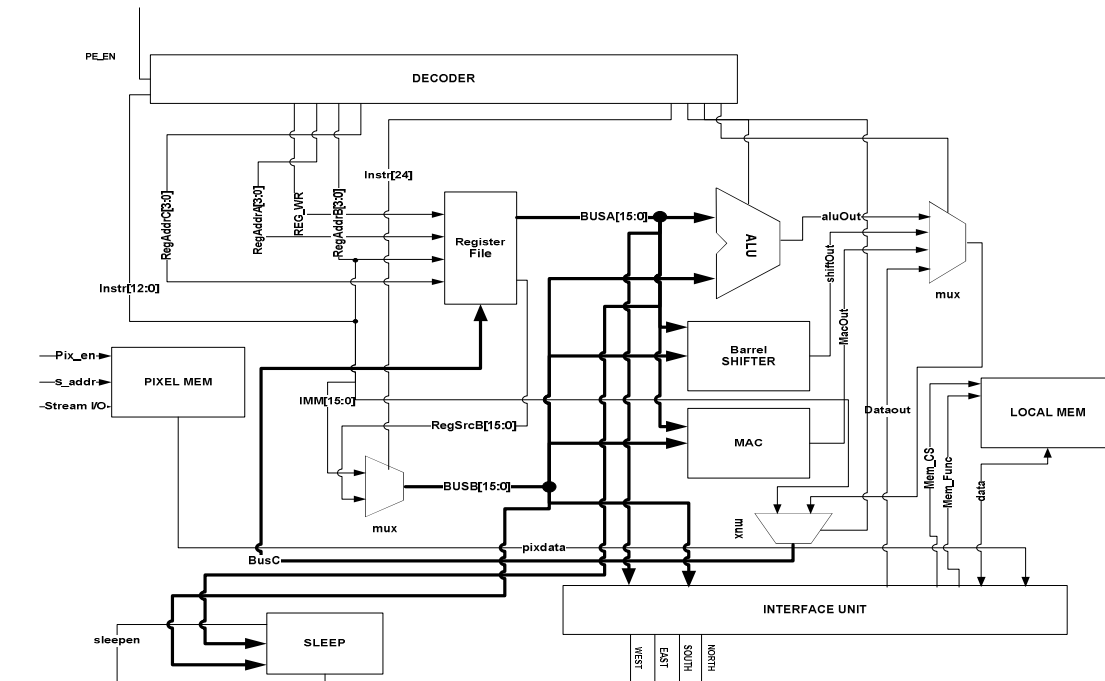


그림 3. 단일 PE 구조
Fig. 3. Single PE architecture

III. 제안하는 SIMD기반 병렬 프로세서 구조

3.1 SIMD기반 병렬 프로세서 전체 구조

제안하는 SIMD기반 병렬 프로세서는 그림 2와 같이 2차원 격자(mesh)구조로 구성된 PE 어레이, 각 데이터의 입출력을 위한 로컬 메모리 및 각 PE와 입출력 유닛을 제어하는 ACU(Array Control Unit)로 구성되어 있다. PE 어레이는 16개 (4x4)의 PE로 구성되어 있다.

ACU는 프로그램 메모리에서 명령어를 패치 (fetch)하여 전체 PE에 동시에 전송 (broadcasting)하며, PE를 활성화 (active) 및 비활성화 (sleep) 시키는 역할을 한다. DEI(data exchange interface)는 외부 I/O와 PE 로컬 레지스터에 데이터를 전송 하는 역할을 한다.

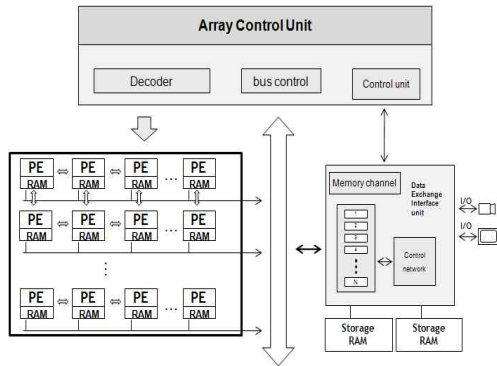


그림 2. SIMD기반 병렬 프로세서 구조
Fig. 2. SIMD based parallel processor architecture

3.2 단일 PE 구조

각 PE는 간단한 RISC 아키텍처로 구성되어 있으며, 연산 및 데이터 처리를 16비트 단위로 수행한다. 그림 3은 단일 프로세싱 엘리먼트 (PE) 구조를 보여준다. 각 PE는 8192개의 로컬 메모리, 산술 논리 장치(ALU), 배럴 시프트 (barrel shift), 곱셈 및 누산기(multiply-accumulate, MAC) 등을 포함한다. 또한 지역 데이터의 정보에 의해 각 PE를 활성화/비활성화 시키는 sleep장치, 이웃하는 PE들과 데이터 통신을 위한 NEWS (North-East-West-South) 네트워크 및 외부 데이터와 통신을 수행하는 serial I/O유닛을 포함한다. 레지스터 파일은 16개로 구성되어 있으며, 2개의 읽기 포트 (BusA, BusB)와 1개의 쓰기 포트 (BusC)를 가지고 있다. 디코더는 읽어온 명령을 해석하고 PE_EN 입력 신호가 활성화

화되면 PE에서 프로그램을 실행할 준비를 한다.

각 PE의 곱셈 및 누산기는 32bit 크기의 누산기로 구성되어 있으며 연산은 싱글 사이클로 동작한다. 하지만, 상위 16비트와 하위 16비트를 레지스터 파일로 저장할 때에는 1 사이클이 더 소요된다. 산술 및 논리 연산 (ALU) 유닛은 논리 곱(AND), 논리합(OR), 배타적 논리합(XOR)을 위한 논리 모듈과 덧셈과 뺄셈을 위한 산술 모듈로 구성되어 있으며, 배럴 쉬프트 모듈은 32비트 범위까지 좌/우로 쉬프트 가능하다. 활성 및 비활성 유닛은 각 PE의 지역 정보 상태 및 조건에 따라 활성화 및 비활성화 상태를 만든다. 프로그램 수행중 PE의 비활성화 상태가 많아지면, 전체적으로 PE 이용률 (utilization)이 저하되어 성능에 영향을 미치게 된다. 이러한 문제점을 해결하기 위해서는 효율적인 알고리즘 구현이 요구된다. 각 PE간의 데이터 통신은 NEWS(North-East-West-South) 커뮤니케이션 유닛이 담당하며, 싱글 사이클로 데이터 전송이 가능하도록 설계되었다. 로컬 I/O 메모리는 데이터의 입력과 출력이 같은 저장장소를 가지는 구조로 설계되었으며, DEI는 데이터 I/O의 입출력 수행을 담당한다.

3.3 데이터 교환 인터페이스

DEI는 외부 I/O 컨트롤러 및 메모리와 PE의 로컬 메모리간의 데이터 통신을 담당한다. 그림 4는 PE와 DEI사이의 연결 인터페이스를 보여준다. Arbiter는 PE의 지역 메모리에 접근하기 위한 권한을 부여한다.

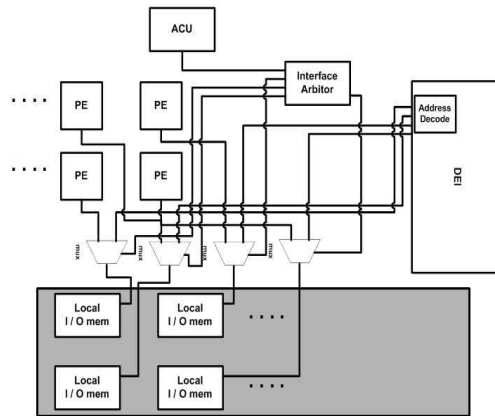


그림 4. DEI 와 PE의 연결 인터페이스
Fig. 4. Connection interface between DEI and PE

IV. 기능적 특징

4.1 파이프 라이닝

그림 5에서 보느냐와 같이 SIMD기반 병렬 프로세서는 페치(Fetch), 디코더(Decode), 실행(Execution)의 3단계 파이프라인 구조로 설계되었다. 1단계 (페치)에서는 ACU가 명령어 메모리로부터 명령어(instruction)을 페치한다. 2단계 (디코더)에서는 ACU의 디코더 유닛이 ACU에서 수행되는 스칼라(Scalar)명령어인지 PE에서 수행되는 벡터(vector) 명령어인지를 구분하여 BusA, BusB, BusC 각 포트에 해당되는 레지스터 주소 및 immediate값을 할당한다. 마지막 3 단계 (실행)에서는 각 유닛들이 명령어와 컨트롤 시그널에 의해 실행된다.

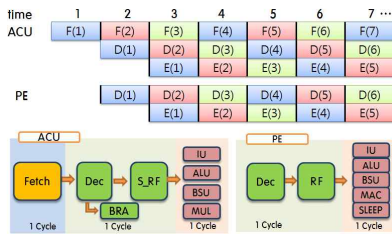


그림 5. 병렬프로세서의 파이프라이닝
Fig. 5. Pipelining of a parallel processor

4.2 PE의 활성화

본 논문에서 제안하는 병렬프로세서의 각 PE는 레지스터의 지역 데이터 조건에 따라 활성화 및 비활성화 상태가 결정된다. 활성화/비활성화 상태를 결정하는 플래그에는 3가지가 있다. PE내부의 레지스터 값이 0인지를 판단하는 제로 플래그, 부호를 판단하는 부호 플래그, 16비트 레지스터로 구성된

sleep 벡터 플래그를 포함하고 있다. 부호 플래그와 제로 플래그는 레지스터의 값에 따라서 업데이트 되고 그에 따라 활성화 및 비활성화 상태가 결정되는 반면, 벡터 플래그는 읽기(READ)포트에 의해 전달되는 신호를 해당되는 비트에 set을 시켜서 비활성화 상태를 만든다. 다시 활성화 상태를 만들려면 벡터 플래그의 set을 clear시키면 된다.

그림 6은 레지스터 값의 조건에 따라 PE 중 일부가 Sleep 명령어에 의해 비활성화 되는 상태를 보여준다. 붉은색의 PE들은 레지스터 R1 값이 음수이므로 비활성화 되어 Wakeup 명령어를 만날 때까지 명령어들을 수행하지 않는다. 반면, 초록색의 PE들은 활성화 상태로써 이어지는 명령어들을 수행한다. 따라서 이러한 Sleep과 Wakeup 명령어를 통해 조건부 PE의 부분적 실행을 수행할 수 있다.

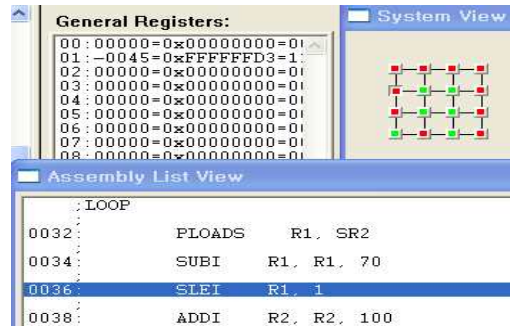


그림 6. Sleep 명령어를 이용한 PE 비활성화
Fig. 6. Deactivation of PEs using a sleep instruction

표 2. 평가 지표 요약
Table 2. Summary of evaluation metrics

실행시간 (execution time)	처리량 (sustained throughput)	에너지 효율 (energy efficiency)
$t_{exec} = \frac{C}{f_{ck}}$	$Th_{sust} = \frac{O_{exec} \cdot U \cdot N_{PE}}{t_{exec}}$	$\eta_E = \frac{O_{exec} \cdot U \cdot N_{PE}}{Energy} [\frac{Gops}{Joule}]$
<p>C: 사이클 개수, f_{ck}: 클럭 주파수, O_{exec}: 수행된 연산 개수 U: 프로세싱 엘리먼트 이용률, N_{PE}: 프로세싱 엘리먼트의 개수</p>		

표 3. 16개 프로세싱 엘리먼트 시스템에서의 전체성능
Table 3. Overall performance of 16 PEs system

알고리즘	Total cycle [cycles]	Vector instruction	Scalar instruction	System utilization [%]	Sustained throughput [Gops/sec]	Execution time [ms]
Median Filter	991437	733190	258247	85.79	7.3090	1.38
DCT	194147	169690	24457	99	10.068	0.27
DWT	1326265	1085527	240738	67.60	6.37	1.84

V. 성능평가

5.1 실험 방법론

세 가지 레벨 (애플리케이션, 아키텍처, 테크놀로지)로 구성되어 있는 SIMD기반 병렬 프로세서의 실험 방법론은 그림 7과 같다. 애플리케이션 레벨에서는 명령어 레벨의 SIMD기반 병렬프로세서용 정밀 사이클 시뮬레이터를 이용하여 사이클 개수, 동적 명령어 빈도, 프로세싱 엘리먼트 이용률 (utilization) 등의 실행 데이터를 추출한다. 아키텍처 레벨에서는 모델링된 아키텍처의 디자인 변수들을 계산하기 위해 Chai [17]가 제안한 SIMD 병렬 프로세서용 이중 아키텍처 모델링 툴을 사용하였다. 테크놀로지 레벨에서는 각 아키텍처 모델들의 테크놀로지 변수 (latency, power, clock frequency)를 계산하기 위해 Generic System Simulator (GENESYS)를 사용하였다 [18]. 마지막으로 세 레벨에서 구해진 데이터베이스를 조합하여 각 경우에 대한 실행시간, 처리량, 에너지 효율을 결정하였다.

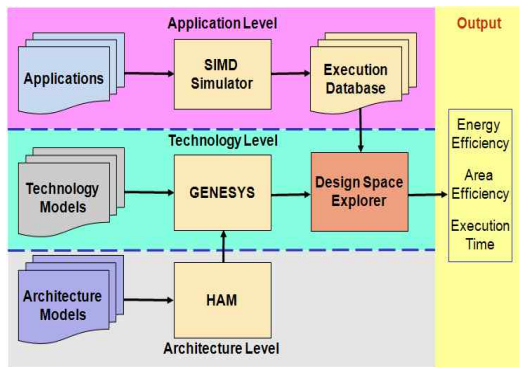


그림 7. SIMD 기반 병렬 프로세서 시뮬레이션을 위한 실험방법론
Fig. 7. Experiment methodology for SIMD parallel processor simulation

5.2 성능 분석 지표

제안하는 병렬 프로세서의 성능을 평가하기 위해 선택된 멀티미디어 애플리케이션을 SIMD기반 병렬프로세서용 정밀 사이클 (cycle-accurate) 시뮬레이터를 사용하여 구현하였으며, 아키텍처 파라미터는 표 1과 같다. 제안하는 병렬프로세서는 16개의 PE를 가지고 있으며 격자 (mesh)구조로 연결되어 있다. 각 프로세싱 엘리먼트는 이미지 사이즈와 PE의 개수를 고려하여 8192개의 16비트 메모리를 사용하여 시뮬

레이션 하였으며, 130nm 테크놀로지와 100MHz 및 720MHz 클럭주파수를 이용하여 성능을 분석하였다.

표 1. 제안하는 병렬프로세서 아키텍처 파라미터
Table 1. Parameters of the proposed parallel processor architecture

parameter	value
Number of PEs	16
Memory/PE [16bit - word]	8196
VLSI Technology	130nm
Clock Frequency	100MHz (720MHz)
intALU/intMUL/Barrel Shifter/ intMACC/comm	1 / 1 / 1 / 1 / 1

표 2는 제안하는 병렬 프로세서의 성능을 평가하기 위한 지표를 보여 준다 [19]. 실행 시간 (execution time)은 알고리즘이 수행되는 시간을, 처리량 (sustained throughput)은 단위 시간당 처리되는 명령어 개수 (Giga-operations/second)를, 에너지 효율 (energy efficiency)은 단위 에너지 당 소비된 명령어 개수 (Giga-operations/Joule)를 나타낸다. 또한 기존의 병렬프로세서와의 성능을 비교하기 위하여 프로세싱 엘리먼트 당 상대 연산 처리량 (competitive computation throughput/PE)을 사용하였다.

5.3 성능 평가

100MHz 클럭 주파수에서 동작하는 16개의 프로세싱 엘리먼트 시스템을 이용하여 이산 웨이블릿 변환 (DWT), 미디언 필터링 (Median Filtering) 및 이산 코사인 변환 (DCT)을 수행한 결과에 대한 전체 성능은 표 3과 같다. 이들 알고리즘들은 제안한 병렬프로세서에서 실시간 (<30ms at 30fps)으로 처리 가능함을 알 수 있다.

표 4는 멀티미디어 애플리케이션에 대한 제안한 SIMD기반 병렬프로세서와 기존의 병렬프로세서들 간의 성능비교를 보여준다. 제안한 병렬프로세서는 기존의 병렬프로세서보다 프로세싱 엘리먼트 당 상대 연산 처리량 (pixels/(sec * # of PE))에서 높은 성능을 보인다.

기존의 병렬프로세서보다 높은 성능을 보인 주요 원인을 살펴보면 다음과 같다. 첫 번째, 제안한 SIMD기반 병렬프로세서는 MAC (Multiply-Accumulate)유닛의 지원으로 기존의 몇몇 프로세서보다 애플리케이션의 성능 향상에 이점이 있다. DCT와 DWT 응용프로그램의 경우, MAC 연산이 주를 이루기 때문에 이러한 응용 프로그램에서는 높은 성능 향

표 4. 제안하는 병렬프로세서와 기존 병렬 프로세서의 성능 비교 [10],[11],[12],[13],[14]
 Table 4. Performance comparison of the proposed parallel processor and conventional parallel processors

	DWT			Median Filter				DCT			
	Maspar1	Maspar2	Proposed	MGAP	Slim	iCPA	Proposed	MGAP	Kestrel	iCPA	Proposed
Number of PEs	16,384	16,384	16	16,385	5,000	64	16	16,385	64	64	16
Clock Frequency [Mhz]	12	12.5	100	25	25	100	100	25	33	100	100
Image size	512 X 512	512 X 512	256 X 256	128 X 128	100 X 50	640 X 480	256 X 256	128 X 128	256 X 256	640 X 480	256 X 256
Execution Time(ms)	5,750	15	13.26	0.013	0.009	12.29	9.91	8.637	368.7	41.02	0.27
Performance (Mpixels/sec)	0.06	17.47	4.94	1,260	556	24.9	6.66	2	0.2	7	33.7
Performance [pixels/sec * # of PE]	0.3052	1066.3	308,750	76.9	111.2	389	416.25	0.122	15.63	125	2106.3

상을 보여준다. 표 5는 제안한 프로세서와 기존의 프로세서에서 MAC 유닛의 지원 유무를 보여준다.

표 5. MAC 유닛을 포함하는 병렬프로세서
 Table 5. Parallel Processors including a MAC unit

	Kestrel	MasPar2	MGAP	Slim	iCPA	proposed
MAC Unit	Yes	Yes	No	No	No	Yes

두 번째, 제안한 병렬프로세서를 이용하여 애플리케이션을 효율적으로 구현함으로써 성능을 향상시킬 수 있다. 예를 들어, DWT (Discrete Wavelet Transform) 응용프로그램을 제안하는 병렬프로세서를 이용하여 구현하면 다음과 같다.

DWT 응용 프로그램은 크게 lowpass와 highpass filter 적용 부분과 filter의 결과 데이터를 재배열하는 부분으로 구성되고, 그림 8은 첫 번째 스테이지 (stage)의 웨이블릿 분해 과정과 lowpass filter의 결과 데이터 E를 재배열하는 과정을 보여준다. 각 PE들은 픽셀 데이터 s, lowpass filter h 및 highpass filter g를 이용하여 컨볼루션 (convolution) 및 다운 샘플링하여 E와 F의 결과를 도출한다. 그림 8-a)는 각 PE 내부에 적재되어 있는 E와 F의 결과를 보여준다. 그림 8-b)는 PE의 column ID를 계산하

여 서쪽 방향의 이웃 PE들에게 E 결과 데이터를 동시에 전달함으로써 제일 왼쪽 column PE들은 E의 column 1, 2의 결과를 가지게 된다. 그림 8-c)는 제일 왼쪽 column PE들을 비활성화 시키고 나머지 PE들은 다시 서쪽 방향의 이웃 PE에게 데이터를 전달함으로써 제일 왼쪽 다음 column PE들이 E의 column 3, 4의 결과를 가지므로써 E (lowpass filter 1,2,3,4)의 결과가 재배열된다. 그림 8-d)는 F 데이터 (highpass filter 1,2,3,4)를 위와 동일한 과정으로 서쪽이 아닌 동쪽으로 이웃 PE들에게 전달함으로써 최종적으로 E와 F의 재배열된 결과를 보여준다. 열 (row)부분 역시 행과 열을 바꾸어 수행하면 첫 번째 스테이지의 웨이블릿 분해과정이 완성된다. 위와 같이 4번의 반복 수행을 통해 4 stage 분해 과정을 효율적으로 구현할 수 있다.

세 번째, 이웃하는 PE들과의 통신 (communication) 횟수 및 소요되는 사이클 수가 적기 때문에 성능 측면에서 이점이 있다. 제안한 병렬 프로세서는 16개의 PE로 구성되어 있고, 각 PE는 균등하게 분배된 충분한 데이터를 로컬 메모리에 저장함으로써 참조 데이터의 재배열 과정에서 요구되는 PE 간 데이터 통신 명령어의 횟수를 줄일 수 있다. 또한, Maspar2 프로세서의 경우, x-net의 근접 연결망을 이용하여 근접통신을 수행하는 경우, x-net구조의 초기화를 위해 3개의 명령어 사이클이 소요되기 때문에 근접 통신에 많은 오버헤드가 발생한다.

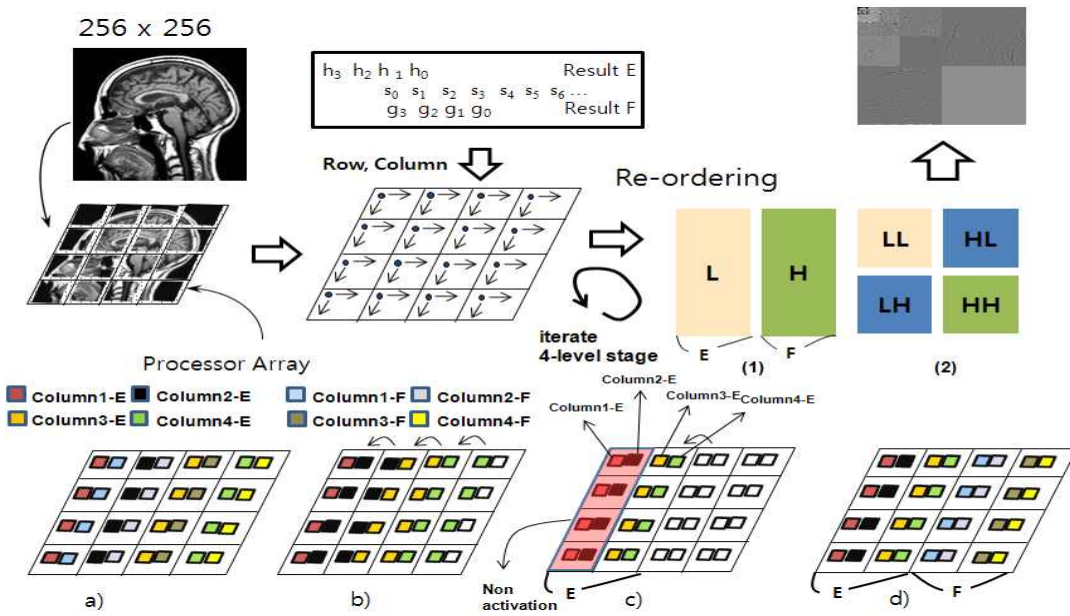


그림 8. 제안한 SIMD 기반 병렬 프로세서를 이용한 DWT 구현 과정
 Fig. 8. Implementation process of DWT using the proposed SIMD-based parallel processor

표 6. 제안하는 병렬 프로세서와 TI C6416의 성능 비교

Table 6. Performance comparison between the proposed parallel processor and TI C6416

Algorithm		DCT		Median Filter		DWT	
parameter	unit	병렬프로세서	TI C6416	병렬프로세서	TI C6416	병렬프로세서	TI C6416
Technology	[nm]	130	130	130	130	130	130
Clock Frequency	[Mhz]	720	720	720	720	720	720
Average Power	[mW]	2,003.9	950	1,803.97	950	1420	950
Average Throughput	[MIPS]	10,068	790	7,309.04	527.699	6,373.4	717.624
Execution Time	[ms]	0.269	0.37	1.3769	37.3	1.84	57.8
Energy	[μjoule]	540.42	351.5	2484.1	35435	2615.65	54910
Energy Efficiency	[Gops/Joule]	5.024	0.8315	4.051	0.5	4.488	0.755

5.4 상용 DSP와의 성능 비교

또한 본 논문은 제안한 병렬프로세서와 기존의 상용 고성능 DSP인 TI C6416 [20]를 동일한 130nm 테크놀로지와 720MHz의 클럭 주파수에서 성능을 비교하였다. 제안한 병렬프로세서는 16개의 프로세싱 엘리먼트(PE)를 사용하여 데이터 레벨 병렬성(data-level parallelism)을 지향하는 반면, TI C6416은 8-way VLIW (Very Long Instruction Word) 구조로서 8개의 명령어를 동시에 수행 가능한 명령어 레벨 병렬성(instruction-level parallelism)을 추구한다.

표 6은 제안하는 병렬프로세서와 고성능 TI C6416 DSP와의 성능 비교를 보여 준다. 제안한 병렬프로세서는 TI C6416보다 1.4~31.4배의 성능 향상 및 5.9~8.1배의 에너지 효율 향상을 보인다. 그림 9과 10은 제안한 병렬프로세서를 TI C6416과 비교하여 실행시간 및 에너지 효율 측면에서 우수성을 막대그래프로 보여 준다. 이러한 결과는 제안하는 SIMD 기반 병렬프로세서의 잠재가능성을 보여준다.

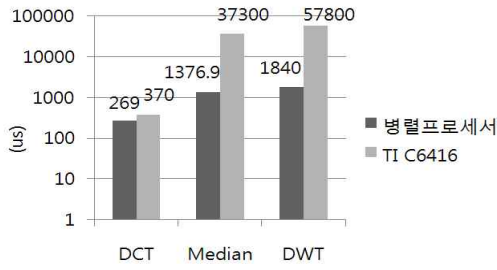


그림 9. 실행시간 비교
Fig. 9. Execution time comparison

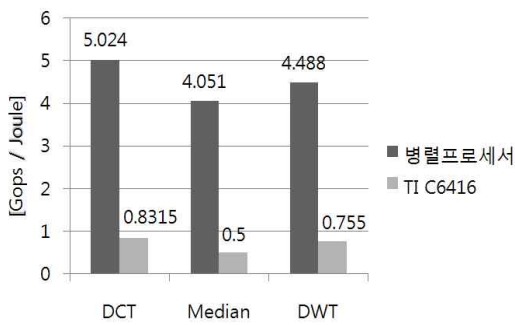


그림 10. 에너지 효율 비교
Fig. 10. Energy efficiency comparison

VI. 합성 결과

제안한 SIMD기반 병렬 프로세서를 RTL레벨 [21]에서 설계 및 합성하였고, Xilinx사의 Vertex-4 XC4VLX60 FPGA [22] 프로토타입 시스템을 이용하여 기능 검증하였다. 그림 11은 16개의 PE를 내장한 병렬 프로세서의 부분적인 합성 스키마틱을 보여주며, 합성한 결과는 표 7과 같다. ACU는 1,021개의 LUT (Look-Up Table)가 사용되었으며, 각 PE는 681개의 LUT와 8K byte RAM이 사용되었다. 또한, DEI (data exchange interface)는 1,010개의 LUT와 2K byte RAM이 사용되었다. 전체적으로 16 PE로 구성된 병렬 프로세서는 11,972 LUT와 136K byte의 RAM이 사용되었다.

표 7. 16개의 PE 합성 결과
Table 7. Synthesis result of 16 PEs

	LUT	RAM(Kbyte)
ACU	1,021	-
Each PE	681	8
DEI	1,010	2
Total (16 PEs)	11,972	136

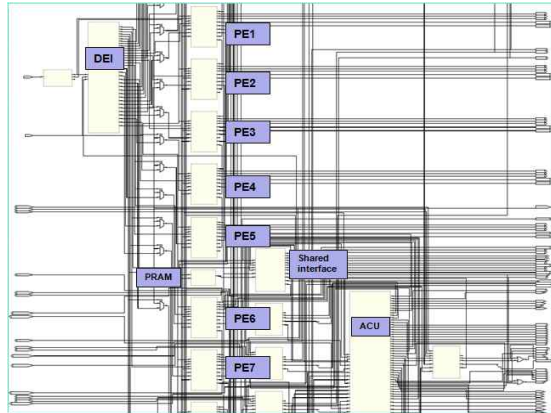


그림 11. 병렬프로세서의 합성 스키마틱
Fig. 11. Synthesis schematic of the parallel processor

VII. 결론

본 논문에서는 멀티미디어의 효율적인 처리를 위해 SIMD 기반 병렬 프로세서를 제안하였다. 제안한 병렬프로세서는 16개의 프로세싱 엘리먼트를 포함하고 있으며, 각 프로세싱 엘리먼트는 메쉬 구조의 상호연결망으로 연결되어 있다. 또한, 각 프로세싱 엘리먼트는 3단계 파이프라인 (패치, 디코더, 실행)으로 설계되었다. 모의실험 결과, 제안한 병렬프로세서는 기존의 병렬프로세서보다 높은 연산 처리량을 보였다. 또한 상용 고성능 TI DSP와의 비교에서도 높은 성능 및 에너지 효율 향상을 보였다. 이러한 결과를 통해 제안한 SIMD 기반 병렬프로세서의 무한한 잠재가능성을 알 수 있다. 제안한 병렬프로세서를 RTL레벨에서 설계 및 합성하였고, Xilinx사의 FPGA 프로토타입 시스템을 이용하여 기능을 검증하였다.

참고 문헌

- [1] M.K. Chung, S.M. Park, and N.W. Eum, "Technology and trend of parallel processor," Electronics and Telecommunications Trends, vol. 24, no. 6, pp. 86-93, Dec. 2009.
- [2] S.H. Lee, "The design and implementation of parallel processing system using Nios(R) II embedded

- processor”, *Journal of the Korea Society of Computer and Information*, vol. 14, no. 1, pp. 97-103, Nov. 2009.
- [3] J.J. Lee, S.M. Park, and N.W. Eun, “Application specific instruction set processor for multimedia applications,” *Electronics and Telecommunications Trends*, vol. 24, no. 6, pp. 94-98, Dec. 2009.
- [4] S.H. Kim, S.B. Nam, and H.J. Lim, “An improved area edge detection for real-time image processing,” *Journal of the Korea Society of Computer and Information*, vol. 14, no. 1, pp. 99-106, Jan. 2009.
- [5] P. Ranganathan, S. Adve, and N. P. Jouppi, “Performance of image and video processing with general-purpose processors and media ISA extensions,” in *Proc. of the 28th Intl. Sym on Computer Architecture*, pp. 124-135, May. 1999.
- [6] R. Bhargava, L. John, B. Evans, and R. Radhakrishnan, “Evaluating MMX technology using DSP and multimedia applications,” in *Proc. of IEEE/ACM Sym on Microarchitecture*, pp. 37-46, 1998.
- [7] N. Slingerland and A. J. Smith, “Measuring the performance of multimedia instruction sets,” *IEEE Trans. on Computers*, vol. 51, no. 11, pp. 1317-1332, Nov. 2002.
- [8] A. Shahbahrani, B. Juurlink, and S. Vassiliadis, “Versatility of extended subwords and the matrix register file,” *ACM Transactions on Architecture and Code Optimization (TACO)*, vol. 5, no. 1, Article 5:1-5:30, 2008.
- [9] J.-C. Chiu, Y.-L. Chou, and H.-Y. Tzeng, “A Multi-streaming SIMD Architecture for Multimedia Applications”, in *Proceedings of the 6th ACM conference on Computing frontiers*, pp. 51-60, 2009.
- [10] A. Gentile and D. S. Wills, “Portable video supercomputing,” *IEEE Trans. on Computers*, vol. 53, no. 8, pp. 960-973, Aug. 2004.
- [11] A. Krikelis, I. P. Jalowiecki, D. Bean, R. Bishop, M. Facey, D. Boughton, S. Murphy, and M. Whitaker, “A programmable processor with 4096 processing units for media applications,” in *Proc. of the IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 937-940, May. 2001.
- [12] L. W. Tucker and G. G. Robertson, “Architecture and applications of the connection machine,” *IEEE Computer*, vol. 21, no. 8, pp. 26-38, 1988.
- [13] “Connection machine model CM-2 technical summary,” Thinking Machines Corp., version 51, May 1989.
- [14] MarPar (MP-2) System Data Sheet. MarPar Corporation, 1993.
- [15] M. J. Irwin, R. M. Owens, “A two-dimensional, distributed logic processor,” *IEEE Trans. on Computers*, vol. 40, no. 10, pp. 1094-1101, 1991.
- [16] M. Bolotski, R. Armithrajah, W. Chen, “ABACUS: A high performance architecture for vision,” in *Proceedings of the International Conference on Pattern Recognition*, 1994.
- [17] S. M. Chai, T. Taha, D. S. Wills, J. D. Meindl, “Heterogeneous architecture models for interconnect-motivated system design,” *IEEE Trans. on VLSI Systems*, vol. 8, no. 6, pp. 660-670, 2000.
- [18] J. C. Eble, V. K. De, D. S. Wills, J. D. Meindl, “Generic system simulator (GENESYS) for ASIC technology and architecture beyond 2001,” *Proceedings of the Ninth Annual IEEE International ASIC Conference*, pp. 193-196, September 1996.
- [19] Jongmyon Kim, Yong-Min Kim, Cheol-Hong Kim, “Performance evaluation of multimedia extensions on variable many-core processors,” in the *International Conference on Computer Design (CDES’10)*, pp. 98-104, July 2010.
- [20] TMS320C64x families: http://www.bdti.com/procs_um/tic64xx.htm
- [21] M.-K. You, Y.-J. Oh, G.-Y. Song, “System-level hardware function verification system,” *Journal of The Institute of Signal Processing and Systems*, vol. 11, no. 2, pp. 86-91, April 2010.
- [22] Xilinx Userguide: http://www.xilinx.com/support/documentation.user_guides/ug070.pdf

저 자 소 개



김 용 민

2009: 울산대학교 컴퓨터공학사.
 2009: 울산대학교 컴퓨터정보통신공학부 석사과정 입학.
 관심분야: 임베디드 시스템, 컴퓨터구조, 병렬처리

Email: jafstar@nate.com



황 철 희

2008: 울산대학교 컴퓨터공학사.
 2008: 울산대학교 컴퓨터정보통신공학부 석사과정 입학.
 관심분야: 임베디드 SoC, 컴퓨터구조, 고장예측, 병렬처리

Email: imus@hanmail.net



김 철 홍

1998 : 서울대학교 컴퓨터공학사.
 2000 : 서울대학교 컴퓨터공학부 석사.
 2006 : 서울대학교 전기컴퓨터공학부 박사
 2005 - 2007년 : 삼성전사 반도체총괄 책임연구원
 2007 - 현재 : 전남대학교 전자컴퓨터공학부 교수
 관심분야 : 임베디드시스템, 컴퓨터구조, SoC설계, 저전력 설계

Email: cheolhong@gmail.com



김 종 면

1995 : 명지대학교 전기공학사
 2000 : University of Florida ECE 석사
 2005 : Georgia Institute of Technology ECE 박사
 2005 - 2007 : 삼성종합기술원 전문연구원
 2007 - 현재 : 울산대학교 컴퓨터정보통신공학부 교수
 관심분야 : 프로세서 설계, 임베디드 SoC, 컴퓨터구조, 병렬처리

Email: jongmyon.kim@gmail.com