

3차원 구조 멀티코어 프로세서의 분기 예측 기법에 관한 온도 효율성 분석

안진우* · 최홍준** · 김종면*** · 김철홍****

요 약

프로세서의 성능을 효율적으로 증가시키기 위한 기법 중 하나로 명령어 수준의 병렬성을 높이는 추론적 수행(Speculative execution)이 사용되고 있다. 추론적 수행 기법의 효율성을 결정하는 가장 중요한 핵심 요소는 분기 예측기의 정확도이다. 하지만, 높은 예측율을 보장하는 복잡한 구조의 분기 예측기를 최근 주목 받고 있는 3차원 구조 멀티코어 프로세서에 적용하는데 있어서는 발열 현상이 큰 장애요소가 될 것으로 예측된다. 본 논문에서는 3차원 구조 멀티코어 프로세서에서 발생할 수 있는 분기 예측기의 높은 발열 문제를 해결하기 위해 두 가지 기법을 제시하고, 이에 대한 효율성을 상세하게 분석하고자 한다. 첫번째 기법은 분기 예측기의 온도가 임계 온도 이상으로 올라가는 경우 분기 예측기의 동작을 일시적으로 정지시키는 동적 온도 관리 기법이고, 두번째 기법은 3차원 구조 멀티코어 프로세서의 각 층 별로 온도를 고려하여 서로 다른 복잡도를 지닌 분기 예측기를 차등 배치하는 기법이다. 두 가지 기법 중에서 복잡도를 고려한 차등 배치 기법은 평균 87.69°C의 온도를 나타내는 반면, 동적 온도 관리 기법은 평균 89.64°C의 온도를 나타내었다. 그리고, 각 층에서 발생하는 온도 변화율을 각 기법에 대하여 비교한 결과, 동적 온도 관리 기법의 온도 변화율은 평균 17.62°C를 나타내었고 복잡도 차등 배치 기법의 온도 변화율은 평균 11.17°C를 나타내었다. 이러한 온도 분석을 통하여 3차원 멀티코어 프로세서에서 분기 예측기의 온도를 제어하였을 경우, 복잡도 차등 배치 기법을 적용하는 것이 더 효율적임을 알 수 있다. 성능적인 측면을 분석한 결과, 동적 온도 관리 기법은 해당 기법을 적용하지 않았을 경우보다 평균 27.66%의 성능 하락을 나타내었지만, 복잡도 차등 배치 기법은 평균 3.61%의 성능 하락만을 나타내었다.

키워드 : 3차원 구조 멀티코어 프로세서, 분기 예측기, 열섬 현상, 동적 온도 관리 기법

Analysis on the Thermal Efficiency of Branch Prediction Techniques in 3D Multicore Processors

Ahn Jin Woo* · Choi Hong Jun** · Kim Jong Myon*** · Kim Cheol Hong****

ABSTRACT

Speculative execution for improving instruction-level parallelism is widely used in high-performance processors. In the speculative execution technique, the most important factor is the accuracy of branch predictor. Unfortunately, complex branch predictors for improving the accuracy can cause serious thermal problems in 3D multicore processors. Thermal problems have negative impact on the processor performance. This paper analyzes two methods to solve the thermal problems in the branch predictor of 3D multi-core processors. First method is dynamic thermal management which turns off the execution of the branch predictor when the temperature of the branch predictor exceeds the threshold. Second method is thermal-aware branch predictor placement policy by considering each layer's temperature in 3D multi-core processors. According to our evaluation, the branch predictor placement policy shows that average temperature is 87.69°C, and average maximum temperature gradient is 11.17°C. And, dynamic thermal management shows that average temperature is 89.64°C and average maximum temperature gradient is 17.62°C. Proposed branch predictor placement policy has superior thermal efficiency than the dynamic thermal management. In the perspective of performance, the proposed branch predictor placement policy degrades the performance by 3.61%, while the dynamic thermal management degrades the performance by 27.66%.

Keywords : 3D Multi-core Processor, Branch Predictor, Hotspot, Dynamic Thermal Management

* 본 논문은 LG연암문화재단의 교수해외연구지원사업으로 수행한 연구결과임.
† 준 회원 : 전남대학교 전자컴퓨터공학부 석사
‡ 준 회원 : 전남대학교 전자컴퓨터공학부 박사과정
*** 정 회원 : 울산대학교 전기공학부 교수

**** 종신회원 : 전남대학교 전자컴퓨터공학부 교수(교신저자)
논문접수 : 2011년 4월 12일
수정일 : 1차 2011년 6월 30일
심사완료 : 2011년 6월 30일

1. 서 론

최근의 고성능 프로세서들은 명령어 수준의 병렬성을 최대한으로 높이기 위해 분기 예측을 통한 추론적 수행 (Speculative execution) 기법을 일반적으로 사용한다. 추론적 수행 기법의 효율성을 결정하는 가장 중요한 핵심 요소 중 하나는 분기 예측기의 정확도이다. 분기 예측기의 정확도가 높을수록 최소한의 분기 예측 실패를 통한 높은 명령어 병렬성을 유지함으로써 프로세서의 성능 향상을 보장할 수 있다. 정확도가 높은 분기 예측기를 적용하게 되면, 잘못된 분기 예측으로 인한 복구 수행에 소모되는 추가적인 전력 낭비 또한 효율적으로 제어할 수 있게 된다[1]. 이와 같은 이유로, 분기 예측기의 정확도 향상을 위한 많은 연구들이 진행된 바 있다[2-6].

하지만, 복잡한 구조를 통해 높은 정확도의 예측을 제공하는 최신 분기 예측 기법들은 프로세서의 성능을 향상시키는데 있어서는 매우 효과적일 수 있으나, 온도 관리 측면에 있어서는 문제점을 발생시킬 수 있다. 특히 최근에 고성능 프로세서 구현 기법으로서 각광받는 기술인 3차원 적층 기술을 적용하는 경우에는 각 층마다 다이(Die)를 적층시켜 구성하므로 이에 따른 전력 밀도가 증가 하기 때문에[7], 분기 예측기의 온도 문제 또한 프로세서를 설계하는데 있어서 반드시 고려되어야 할 것으로 판단된다.

3차원 적층 기술로 구현되는 3차원 구조 멀티코어 프로세서는 많은 수의 트랜지스터로 구성된 코어를 수직으로 적층시키기 때문에 높은 전력 밀도를 유발한다. 이러한 높은 전력 밀도는 프로세서 내부에서 특정 부분의 발열을 심화시키는 열섬(Hotspot) 현상을 발생시킨다[8]. 열섬 현상은 프로세서 내부에서 발생된 열이 전체적으로 분산되지 못하고 특정 영역에서만 밀집되어 특정 영역의 온도가 주변 영역에 비하여 크게 상승하는 현상으로 이와 같은 현상이 심화된다면 프로세서의 타이밍 오류(Timing error)와 같은 기능상의 문제점을 야기할 수 있다[9]. 즉, 3차원 구조 멀티코어 프로세서에서 온도 문제를 효과적으로 해결하지 못한다면 프로세서의 물리적인 변화와 신뢰성에 큰 문제를 가져오게 된다[10]. 또한 이러한 문제점을 해결하기 위해 많은 냉각비용이 추가적으로 소모됨으로써 기존의 2차원 구조보다 더 많은 냉각 비용이 발생하여 프로세서의 경제성이 떨어지는 문제를 유발할 수 있다.

앞서 기술한 바와 같이, 기존의 2차원 구조 멀티코어 프로세서에 적용되는 고성능 분기 예측기를 3차원 구조 멀티코어 프로세서에 적용할 경우에는 심각한 온도 상승으로 인해 분기 예측기에 문제가 발생할 가능성이 존재한다. [11]에 의하면 2개의 층으로 구성된 3차원 구조 멀티코어 프로세서는 최대 125°C의 매우 높은 온도를 발생시킬 수 있다. 이와 같은 심각한 온도 문제를 발생시키는 3차원 구조에서 분기 예측기와 같이 성능에 많은 영향을 미치는 유닛의 온도 문제를 해결하지 않는다면 전체 프로세서의 성능에 악영향을 끼칠 것이다. 그러므로 본 논문에서는 3차원 구조 멀티코어 프로세서에 적용된 분기 예측기의 안정적인 동작을 위한 2

가지 온도 관리 기법을 제시하고, 효율성을 분석하고자 한다. 첫번째 기법은 분기 예측기의 온도가 일정 온도 이상으로 올라가면 분기 예측기의 동작을 일시적으로 정지시키는 동적 온도 관리 기법이고, 두번째 기법은 3차원 구조 멀티코어 프로세서의 각 층에 따른 온도를 고려하여 분기 예측기의 복잡도를 차등 배치하는 기법이다. 본 논문에서는 두 가지 기법에 따른 온도 및 성능을 비교 분석하고, 분석 결과를 기반으로 3차원 구조 멀티코어 프로세서에서 분기 예측기에 발생하는 온도 문제를 효율적으로 해결할 수 있는 기법을 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구 내용을 기술하고, 3장에서는 본 논문에서 제시하는 분기 예측기 차등 배치 기법과 동적 온도 관리 기법에 대하여 기술한다. 4장에서는 실험 환경에 대하여 설명하고, 5장에서는 실험 결과를 토대로 3차원 구조 멀티코어 프로세서에 가장 효율적인 분기 예측기 구조에 대해 살펴본다. 마지막으로, 6장에서 결론을 맺는다.

2. 관련 연구

2.1 동적 온도 관리 기법

공정 기술의 발전은 단일 프로세서 내에 더욱 많은 수의 트랜지스터를 적재시킬 수 있도록 함으로써 프로세서의 성능을 크게 향상시켰다. 그러나 단일 면적 내에 많은 수의 트랜지스터가 밀집됨에 따라 기존의 프로세서와 비교하여 더욱 높은 전력 밀도를 유발하게 되었고 이러한 전력 밀도의 증가 현상은 프로세서의 특정 부분에서 발열이 집중되는 열섬 현상을 야기하였다. 이전에도 온도는 프로세서 설계 시 고려해야 할 사항이었지만, 최근 들어 더욱 중요한 설계 요소로 부각되고 있다. 프로세서의 온도를 낮춰주는 기법은 크게 기계적인 기법과 프로세서의 내부 동작을 제어하는 기법이 있다.

기계적인 기법으로는 프로세서의 열을 방열 판(Heat sink)을 사용하여 분산시키고, 방열 판의 온도 분산 효과를 더욱 강화시키기 위해 방열 판 위에 추가적으로 냉각 팬(Cooling fan)을 장착하는 기법을 들 수 있다. 또한 더욱 더 강력한 온도 제어 효과를 위해 물과 같은 액체를 저온으로 냉각시킨 뒤 이를 이용하여 프로세서의 온도를 제어하는 수냉식 쿨러를 사용하는 기법도 있다.

기계적인 기법과 달리, 프로세서의 내부 동작을 제어함으로써 온도를 제어 하는 동적 온도 관리 기법 또한 널리 사용되고 있다. 특히, 기계적 냉각 기법들이 높은 비용에 비해 낮은 냉각 효율성을 보임에 따라 최근의 온도 관리 연구들은 대부분 프로세서 내부의 동작을 제어하여 발열의 근본 원인을 해결하는 동적 온도 관리 기법에 집중되고 있다. 온도 문제를 해결하기 위해 기존의 2차원 구조 프로세서에서 사용되던 대표적인 동적 온도 관리 기법들에는 공급 전압을 조절하는 가변 전압 조절 기법(DVS: Dynamic Voltage Scaling)과 동작 주파수를 조절하는 가변 주파수 조절 기법(DFS: Dynamic Frequency Scaling)을 들 수 있다[12]. 명령어의 인

출을 지연시켜 온도를 제어하는 인출 지연(Fetch throttling) 기법과 분기 명령어의 예측 실패에 따른 추가적인 전력 소모를 방지하기 위해 예측 실패한 명령어의 파이프라인 수행을 정지시키는 명령어 인출 엔진 정지 기법(Fetch engine gating policy)[13] 또한 제시되었다. 위와 같은 동적 온도 관리 기법들은 프로세서의 온도를 내부 동작의 변경을 통해 낮추기 때문에 기계적인 기법과 비교하였을 때, 매우 효과적으로 프로세서의 온도를 제어할 수 있지만, 이를 위해서 전체 프로세서의 성능을 저하시키는 단점이 존재한다. 이전에 진행되었던 연구에 따르면, 프로세서에 동적 온도 관리 기법을 적용할 경우, 가변 주파수 조절 기법은 약 3%, 가변 전압 주파수 조절 기법은 약 6%, 인출 지연 기법은 약 8% 정도의 성능 저하를 발생시킨다[14]. 또한, 명령어 인출 엔진 정지 기법이 이상적으로 작동한다고 가정하고 프로세서에 적용하였을 경우, 이에 따른 성능 하락은 약 10% 정도 발생하였다[15]. 본 논문에서 목표로 하는 3차원 구조 멀티코어 프로세서는 90℃ 이상의 높은 온도를 나타낸다[11]. 이러한 상황에서 3차원 구조 멀티코어 프로세서에 효율적인 동적 온도 관리 기법이 적용 되지 않는다면 프로세서의 높은 온도에 의하여 심각한 성능 저하 문제를 겪을 수 있다.

2.2 3차원 구조 멀티코어 프로세서 기법

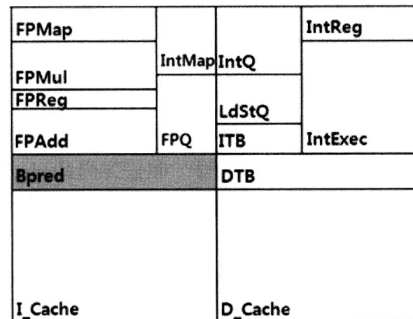
프로세서 제조 공정 기술의 발달과 함께 트랜지스터의 크기가 줄어들면서 단일 프로세서에 여러 개의 코어를 통합시키는 것이 가능해짐에 따라 많은 수의 코어가 적재된 2차원 구조 멀티코어 프로세서가 상용화 되었다. 하지만, 이러한 공정 기술의 발달에 따른 트랜지스터의 크기 감소는 프로세서 내부에 적재되는 코어의 수를 증가시킨 반면, 코어간의 통신을 위해 사용되는 내부 연결망의 길이를 상대적 증가시키게 되었다. 이로 인해 연결망을 통한 코어간 통신의 지연 시간이 증가하여 2차원 구조 멀티코어 프로세서의 성능 향상을 가로막는 문제점이 되었다. 이와 같은 2차원 구조 멀티코어 프로세서의 단점을 극복하기 위하여 최근에 주목을 받고 있는 기법이 3차원 적층 기법을 활용한 3차원 구조 멀티코어 프로세서 기법이다. 3차원 구조 멀티코어 프로세서는 2개 이상의 층을 수직으로 적층시켜 단일 프로세서에 통합시킨다. 3차원 구조 멀티코어 프로세서 기법은 기존의 2차원 구조 멀티코어 프로세서보다 단일 프로세서에 들어갈 수 있는 코어의 수를 증가시키고 코어 간 통신을 위해 사용되는 내부 연결망의 길이를 크게 감소시킬 수 있다. 내부 연결망 길이의 감소는 프로세서의 성능 향상과 함께 연결망에서 소모하는 전력 또한 감소시킨다. 이를 통해 기존의 2차원 구조 멀티코어 프로세서와 비교하여 프로세서의 성능을 크게 향상시킬 수 있을 것으로 기대된다.

3차원 구조 멀티코어 프로세서 기술은 많은 장점들을 가지고 있지만, 단일 프로세서 내에 코어들이 수직으로 배열되기 때문에 프로세서 내의 전력 밀도가 급격하게 증가하여 기존의 2차원 구조 멀티코어 프로세서보다 매우 높은 발열 문제를 발생시키는 단점을 가지고 있다. 이전 연구[16]에 따르면 Alpha 계열의 마이크로프로세서를 이용하여 2개의 층

으로 구성된 3차원 구조 멀티코어 프로세서의 최고 온도를 측정된 결과 기존의 2차원 구조 멀티코어 프로세서 구조와 비교하여 최대 17~20℃ 정도의 온도 상승이 발생한다고 한다. 이는 3차원 구조 멀티코어 프로세서가 매우 심각한 온도 문제를 겪을 수 있음을 나타내는 결과이다. 그러므로 3차원 구조 멀티코어를 설계할 때에는 온도 문제를 우선적으로 해결해야 할 것으로 판단된다.

3. 분석 대상 및 3차원 멀티코어 프로세서의 저온도 분기 예측 기법

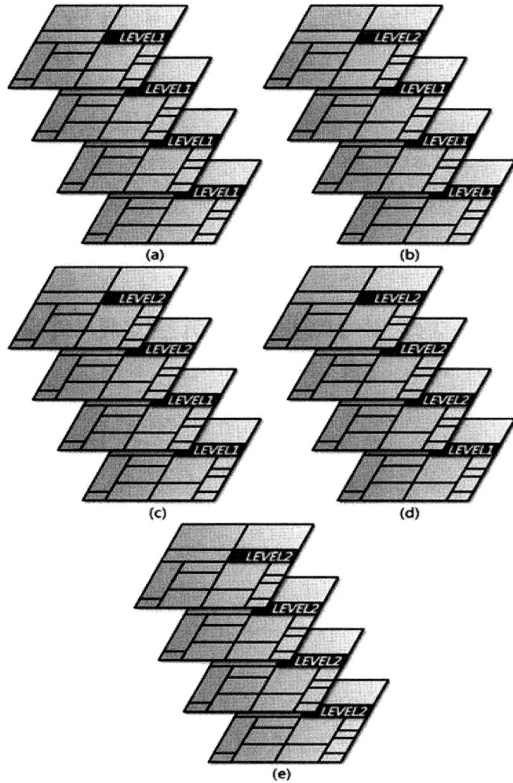
본 논문에서는 3차원 구조 멀티코어 프로세서에서 분기 예측기의 온도 문제를 해결하기 위한 방안으로 동적 온도 관리 기법과 복잡도 차등 배치 기법을 적용하여 각 기법에 따른 온도와 성능을 비교 분석한다. 각 기법들을 분석할 때, 분석 대상으로 3차원 쿼드코어 프로세서를 구성한다. 3차원 쿼드코어 프로세서는 기존의 2차원 구조 쿼드코어 프로세서와 달리 코어들을 각 층마다 수직으로 적층하여 구현된다. 각 층에 배치되는 코어 구조는 Alpha 21264 프로세서를 기반으로 한다.



(그림 1) Alpha 21264 프로세서 코어

(그림 1)은 본 논문의 분석 대상인 3차원 쿼드코어 프로세서에서 사용하는 Alpha 21264 프로세서 코어를 나타내는 것으로 빨간색으로 표시된 부분이 분기 예측기이다. 위의 코어를 사용하여 3차원 쿼드코어 프로세서를 구성할 때, 동일한 방향으로 중첩시켜 총 4개의 층을 이루어 구성된다. 이와 같은 구조에서 분기 예측기의 복잡도 차등 배치를 적용하기 위해, 각 층 별로 배치할 수 있는 분기 예측기의 복잡도를 총 5단계로 구분한다. 각 단계별로 1단계(LEVEL1)가 가장 높은 복잡도를 나타내고 5단계(LEVEL5)가 가장 낮은 복잡도를 나타낸다. 그리고 각 단계에 따른 하드웨어 비용은 4장에 상세히 기술되어 있다.

본 논문에서는 다음과 같은 기준을 토대로 분석 대상인 3차원 쿼드코어 프로세서를 구성한다. 가장 복잡한 구조를 가진 분기 예측기는 온도 문제가 발생할 확률이 상대적으로 높으므로 방열판에서 가장 가까운 층에 배치하여 온도를 최대한 낮추고, 가장 단순한 구조를 가진 예측기는 온도 문제가 발생할 확률이 상대적으로 낮으므로 방열판에서 가장 먼 층



(그림 2) 분석 대상 3차원 쿼드코어 프로세서 구조

에 적재한다. (그림 2)는 분기 예측기의 동적 온도 관리 기법과 복잡도 차등 배치 기법의 분석 대상인 3차원 쿼드코어 프로세서로써 1단계(LEVEL1)와 2단계(LEVEL2)의 복잡도를 나타내는 분기 예측기의 조합으로 구성된다. 총 5가지 구조로 구성되며, 냉각 기능을 수행하는 방열판은 프로세서의 가장 아래에 배치된다. (그림 2)의 구조에서 분기 예측기의 복잡도 차등 배치 기법을 적용하기 위해서는 우선 5가지 구조별로 동적 온도 관리 기법을 적용하지 않았을 경우의 분기 예측기 온도를 분석한 뒤, 높은 온도를 나타내는 분기 예측기를 낮은 복잡도의 분기 예측기로 교체한다. 본 논문의 분석 대상인 3차원 쿼드코어 프로세서에서 분기 예측기를 차등 배치할 때, 다음과 같은 방법으로 온도 구간에 따라 복잡도를 구분하여 배치한다. (그림 2)와 같은 3차원 쿼드코어 프로세서에서 90℃ 미만의 온도를 나타내는 분기 예측기의 경우 복잡도 차등 배치 기법을 적용 하지 않는다. 그러나 90℃ 이상, 95℃ 미만 사이의 온도를 나타내는 분기 예측기의 경우 3단계(LEVEL3)에 해당되는 분기 예측기로 교체한다. 또한, 95℃ 이상, 100℃ 미만 사이의 온도를 나타내는 경우에는 4 단계(LEVEL4)에 해당되는 분기 예측기로 교체한다. 만약 (그림 2)의 (a)구조에서 4층에 적재된 분기 예측기가 100℃ 이상의 온도를 기록하였고 3층에서는 95℃, 2층에서는 90℃를 기록하였다면 4층의 분기 예측기를 5단계로 교체하고 3층과 2층은 각각 4단계와 3단계로 교체하여 구성한다.

본 논문에서는 분석 대상인 3차원 쿼드코어 프로세서에 동적 온도 관리 기법 또한 적용하여 효율성을 분석한다. 3

차원 쿼드코어 프로세서에서 동적 온도 관리 기법을 적용할 경우, (그림 2)의 5가지 구조 별로 각 층마다 파이프라인 수행 중 사용되는 분기 예측기의 온도를 검사하여 임계 온도를 넘어서는 분기 예측기에 대해서는 작동을 일시적으로 중지시킨다.

4. 실험 환경

4.1 모의 실험 환경

본 논문에서는 3차원 쿼드코어 프로세서 구조에 따른 성능 분석을 위해 SPEC2000[17] 벤치마크 중 정수형 벤치마크인 mcf를 SimpleScalar[18] 시뮬레이터를 이용하여 수행하였다. 벤치마크 중 mcf를 선택한 이유는 이전 연구에서[15] 분기 명령어 수행 정지에 따른 프로세서의 성능 저하와 소모 전력 증가가 mcf에서 가장 심각하게 측정되었기 때문이다. 즉, 벤치마크의 특성상 온도 관리를 위해 분기 예측기의 수행을 정지시키는 경우 성능 저하 현상과 함께 소모 전력 증가에 따른 온도 문제 발생 비율이 가장 크게 나타날 것으로 예상되므로 최악의 상황에서 동적 온도 관리 기법이 차등 배치 기법과 비교하여 어느 정도의 효율성을 나타낼 수 있는지를 분석하기 위해 mcf를 선택하였다. 실험에서 사용한 프로세서 구성 변수들은 <표 1>에서 보이는 바와 같다.

<표 1> 프로세서 구성 변수

구성 변수	적용 값	
Functional Units	4 integer ALUs, 4 FP ALUs, 1 integer multiplier/divider, 1 FP multiplier/divider	
Frequency / V _{dd}	3.0GHz/1.5V	
On-chip caches	1	32KB, 4-way, 32bytes lines, 1 cycle latency
	2	512KB, 8-way, 32bytes lines, 12 cycle latency

본 논문에서 적용한 분기 예측기의 복잡도는 <표 2>에서와 같이 5가지 단계로 구분된다. 실험에서 사용된 분기 예측기는 Combined 분기 예측기[19]로써, Bimodal 예측기와 Gshare 예측기 그리고 2개의 예측기 중 1개의 예측기를 선택하는 선택기(Selector)의 조합으로 구성된다.

<표 2> 분기 예측기 구성 변수

단계	Bimodal	Gshare	Selector
LEVEL1	4096B	4096B	4096B
LEVEL2	2048B	2048B	2048B
LEVEL3	1024B	1024B	1024B
LEVEL4	512B	512B	512B
LEVEL5	256B	256B	256B

모의 실험에서는 3차원 쿼드코어 프로세서의 분기 예측기에서 발생하는 온도 문제를 해결하기 위해 각 층 별로 파이

프라인 수행 시 작동하는 분기 예측기의 온도를 검사하여 일정 온도 이상이 되면 일시적으로 동작을 정지시키는 동적 온도 관리 기법을 수행한다. 이 때, 동적 온도 관리 기법이 작동되는 온도를 90℃로 설정하여 해당 온도를 넘어서는 경우에 동작을 정지시켜 온도를 제어한다.

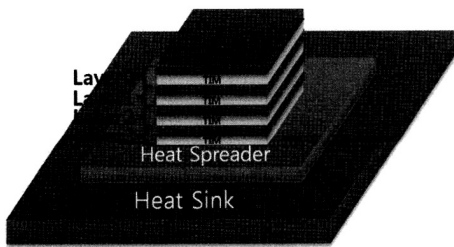
4.2 온도 모델링 환경

본 논문에서는 분석 대상인 3차원 쿼드코어 프로세서에서 각 층 별로 분기 예측기의 온도를 상세하게 분석하기 위해 Hotspot[20]을 이용하여 온도 모델을 설정하였다. 온도 모델을 구성할 때, <표 3>에 나타난 값들을 이용하여 각 층별로 물리적인 특성을 설정하고 이를 토대로 하여 3차원 쿼드코어 프로세서의 온도 값을 측정 하였다.

<표 3> 3차원 쿼드코어 프로세서 온도 모델링

Parameters		Heat capacity (J/m³K)	Resistivity (m-K/W)	Thickness (meter)
Layer0	Die	1.75e6	0.01	0.00015
	TIM	4e6	0.25	2.0e-05
Layer1	Die	1.75e6	0.01	0.00015
	TIM	4e6	0.25	2.0e-05
Layer2	Die	1.75e6	0.01	0.00015
	TIM	4e6	0.25	2.0e-05
Layer3	Die	1.75e6	0.01	0.00015
	TIM	4e6	0.25	2.0e-05

Hotspot에서 설정된 3차원 쿼드코어 프로세서 온도 모델은 방열판이 프로세서 아래에 존재하므로 (그림 3)과 같이 구성된다. 온도 모델링 변수 설정에 의해 방열판과 가장 가까운 코어는 최하층의 Layer3의 다이에 적재되고 가장 멀리 떨어진 코어는 최상층의 Layer0 다이에 적재된다.



(그림 3) Hotspot 온도 모델링

5. 실험 결과

본 논문에서는 3차원 쿼드코어 프로세서를 대상으로 하여 실험을 수행한다. 실험 결과에서 NO-DTM은 3차원 쿼드코어 프로세서에서 동적 온도 관리 기법을 적용하지 않은 경우를 나타내고, DTM은 동적 온도 관리 기법을 적용한 경우를 표시한다. 또한, LEVEL은 3차원 쿼드코어 프로세서에서 분기 예측기의 복잡도 차등 배치 기법을 적용한 경우를 의미한다.

5.1 분기 예측기 조합 결과

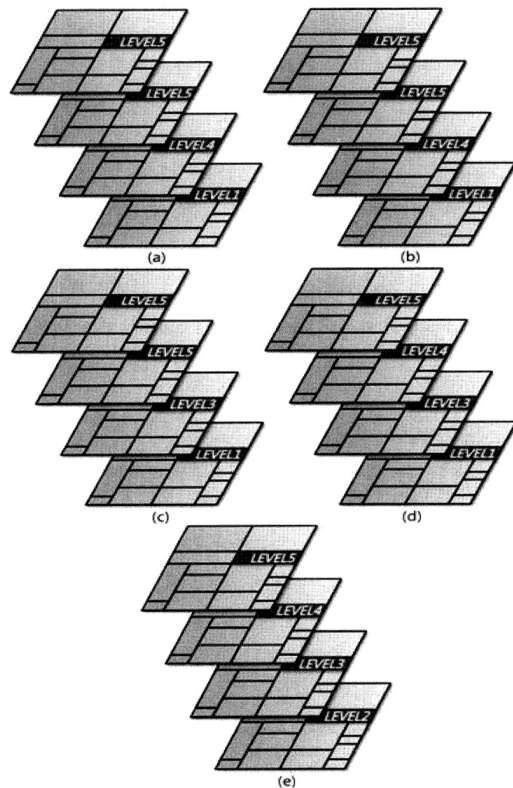
본 논문의 분석 대상 구조인 3차원 쿼드코어 프로세서에서 분기 예측기 복잡도 차등 배치 기법을 적용하기 위해서 우선 (그림 2)의 모든 구조에서 나타나는 분기 예측기의 최고 온도를 분석하였다. 이를 토대로 하여 복잡도 차등 배치 기법에 따라 높은 온도를 나타내는 복잡도가 높은 분기 예측기를 복잡도가 낮은 예측기로 교체하였다.

<표 4> 3차원 쿼드코어 프로세서의 최고 온도

온도(℃)	(a)	(b)	(c)	(d)	(e)
Layer0	107.36	105.12	103.79	103.01	102.66
Layer1	103.68	102.35	100.66	99.72	99.3
Layer2	96.08	95.3	94.36	92.99	92.4
Layer3	83.9	83.51	83.05	82.43	81.46

<표 4>는 (그림 2)의 구조에서 발생한 분기 예측기의 최고 온도를 기록한 것이다. 표 4에 나타난 온도를 토대로 복잡도 차등 배치 기법에 따라 3차원 쿼드코어 프로세서의 분기 예측기를 교체하면 (그림 4)와 같은 새로운 3차원 쿼드코어 프로세서를 구성할 수 있다.

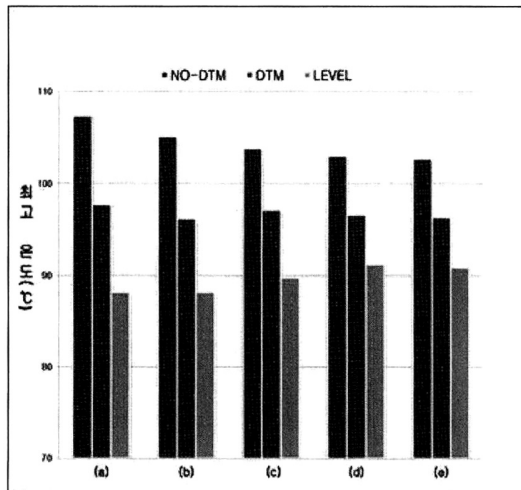
분기 예측기의 복잡도 차등 배치에 의한 새로운 3차원 쿼드코어 프로세서 조합에 있어 항상 다른 조합이 발생하는 것은 아니다. (그림 4)의 (a)와 (b)처럼 분기 예측기의 최고 온도 분포가 유사하여 같은 복잡도 배치 조합을 구성할 수도 있다.



(그림 4) 변경된 3차원 쿼드코어 프로세서

5.2 최고 온도 비교

본 논문에서는 3차원 쿼드코어 프로세서에서 각 기법 별로 가장 높은 온도를 기록한 분기 예측기의 온도를 비교하여 어떠한 기법이 가장 효율적으로 온도를 관리하는지 비교 분석하였다. (그림 5)는 각 기법 별로 3차원 쿼드코어 프로세서에서 나타난 최고 온도 값을 비교한 것이다.

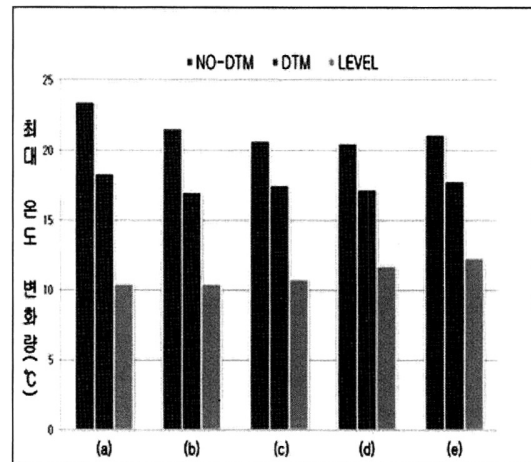


(그림 5) 3차원 쿼드코어 프로세서의 최고 온도 비교

(그림 5)에서 (a)의 경우 동적 온도 관리 기법을 적용하지 않은 경우 최고 온도가 107.36°C이고 동적 온도 관리 기법을 적용하였을 경우 97.72°C, 분기 예측기의 복잡도를 차등 배치한 경우 88.19°C의 온도를 나타낸다. 비교 결과 분기 예측기의 복잡도를 차등 배치한 경우가 동적 온도 관리 기법을 적용한 경우보다 더 낮은 온도를 보임을 알 수 있다. 분기 예측기의 복잡도를 차등 배치한 경우는 평균 89.66°C의 최대 온도를 나타낸다. 이는 동적 온도 관리 기법을 적용하지 않은 경우의 평균 최대 온도인 104.39°C와 동적 온도 관리 기법을 적용한 경우의 평균 최대 온도인 90.19°C를 비교하였을 때 상당히 낮은 온도임을 확인할 수 있다. 즉, 단순한 동적 온도 관리 기법 적용보다 분기 예측기의 복잡도를 차등 배치하는 기법이 더 효율적으로 3차원 쿼드코어 프로세서의 분기 예측기 온도를 조절할 수 있음을 알 수 있다.

5.3 온도 변화량 비교

본 논문에서는 3차원 쿼드코어 프로세서에서 각 기법 별로 분기 예측기의 최대 온도 변화량에 대해 비교 분석을 수행한다. 실험 대상 3차원 쿼드코어 프로세서는 프로세서 구성에 있어서 각 층별로 동일한 플러플랜을 사용한다. 이러한 구조적 특징은 동일한 코어 구성 요소들이 수직 선상에서 위 아래로 중첩되게 한다. 위와 같은 동일한 유닛들간의 중첩된 구조는 전력 밀도를 증가시켜 층 간의 온도문제를 악화시킬 가능성이 크다. 이러한 현상이 심화된다면 3차원 구조 멀티코어 프로세서에서 층 간의 온도 차가 심화되어 온도의 불균등한 분포현상이 나타날 수 있다. (그림 6)은



(그림 6) 3차원 쿼드코어 프로세서의 최대 온도 변화량

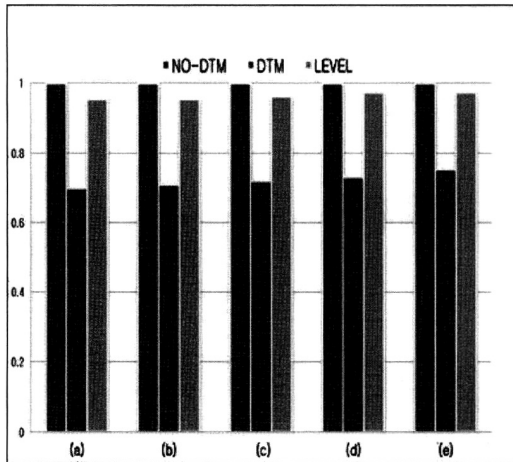
각 기법 별로 3차원 쿼드코어 프로세서에서 나타난 분기 예측기의 온도 변화량을 나타낸 것이다.

(그림 6)에서 (a)의 경우 동적 온도관리 기법을 적용하지 않았을 때 23.46°C의 높은 최대 온도 변화량을 나타내고 동적 온도 관리 기법을 적용한 경우 이보다 낮은 18.38°C를 나타낸다. 그리고 분기 예측기의 복잡도를 차등 배치한 경우에는 10.46°C의 낮은 온도 변화량을 나타낸다. (a)부터 (e)까지 모든 구조에 있어 가장 낮은 최대 온도 변화량을 보인 기법은 분기 예측기의 복잡도를 차등 배치한 경우로 평균 11.17°C의 최대 온도 변화량을 보인다. 이는 동적 온도 관리 기법을 적용하지 않았을 경우 최대 온도 변화량의 평균인 21.52°C, 동적 온도 관리 기법을 적용하였을 때의 최대 온도 변화량의 평균인 17.62°C에 비해 매우 낮은 값을 알 수 있다. 즉 3차원 쿼드코어 프로세서의 분기 예측기 온도 제어에 있어서 가장 효율적인 기법은 분기 예측기의 복잡도를 차등 배치하는 기법임을 알 수 있다.

5.4 성능 비교

본 논문에서는 3차원 쿼드코어 프로세서에서 각 기법 별로 성능 효율성을 측정하여 어떠한 기법이 가장 효율적인지 비교 분석하였다. (그림 7)은 3차원 쿼드코어 프로세서에서 각 기법 별로 성능을 비교한 것이다.

(그림 7)의 결과에서 (a)의 경우 동적 온도 관리 기법을 적용했을 시 동적 온도 관리 기법을 적용하지 않은 경우보다 성능이 약 30% 감소함을 알 수 있다. 이와 달리 복잡도를 차등 배치한 경우는 동적 온도 관리 기법을 적용하지 않은 경우 보다 4.53% 정도로 매우 낮은 성능 감소폭을 보임을 알 수 있다. (b)부터 (e)까지의 구조에 있어서도 동적 온도 관리 기법을 적용한 경우 28.99%, 27.98%, 26.81%, 24.53%의 심각한 성능 감소율을 나타내지만, 분기 예측기의 복잡도를 차등 배치한 경우에는 4.47%, 3.79%, 2.62%, 2.63% 정도의 낮은 성능 감소율을 나타내어 전반적으로 동적 온도 관리 기법을 사용하는 경우보다 분기 예측기의 복잡도를 차등 배치하는 기법이 더 나은 성능을 나타냄을 확



(그림 7) 3차원 쿼드코어 프로세서의 성능 비교

인할 수 있다. 3차원 쿼드코어 프로세서에 분기 예측기의 복잡도를 차등 배치한 기법은 모든 경우에 있어 성능 하락이 발생한다. 그러나 동적 온도 관리 기법을 적용한 기법이 최대 30%의 성능 감소율을 나타내는 것에 비해 복잡도 차등 배치 기법은 이보다 매우 낮은 최대 4.52%의 성능 감소율을 보인다. 이는 분기 예측기의 복잡도 차등 배치 기법이 3차원 쿼드코어 프로세서 구조에서 온도와 성능을 고려할 때 적절한 기법이 될 수 있음을 의미한다.

6. 결론

본 논문에서는 프로세서 성능에 큰 영향을 주는 분기 예측기에 초점을 맞추어, 3차원 구조 멀티코어 프로세서에서 분기 예측기의 온도를 낮출 수 있는 두 가지 기법을 제시하여 효율성을 분석하였다. 첫번째 기법은 분기 예측기의 온도가 특정 임계온도를 넘어서면 작동을 일시적으로 중지시키는 동적 온도 관리 기법이고, 두번째 기법은 분기 예측기의 복잡도를 층별로 차등 배치하여 온도를 낮추는 기법이다. 실험을 수행할 때, 실험 대상인 3차원 쿼드코어 프로세서에 위의 두 가지 기법을 적용하여 실험을 수행하였다. 실험 결과, 복잡도 차등 배치 기법은 평균 87.69°C의 온도를 나타내었고, 동적 온도 관리 기법을 적용한 경우에는 평균 89.64°C의 온도를 나타내었다. 그리고, 각 층에서 발생하는 온도 변화율을 분석한 결과, 동적 온도 관리 기법의 경우 온도 변화율은 평균 17.62°C를 나타내었고 차등 배치 기법의 온도 변화율은 평균 11.17°C를 나타내었다. 이러한 온도 분석을 통하여 분석 대상인 3차원 쿼드코어 프로세서에서 분기 예측기의 온도를 제어할 경우 복잡도 차등 배치 기법이 더 효율적임을 알 수 있다. 분석 대상인 3차원 쿼드코어 프로세서에서 분기 예측기의 성능적인 측면을 분석한 결과, 동적 온도 관리 기법의 경우, 해당 기법을 적용하지 않은 경우보다 평균 27.66%의 성능 하락을 나타내었지만 복잡도 차등 배치 기법은 해당 기법을 적용하지 않은 경우보다 평

균 3.61%의 낮은 성능 하락만을 나타내었다. 위와 같은 결과들은 3차원 구조 멀티코어 프로세서에서 분기 예측기의 온도를 제어할 때, 온도 효율성을 고려하여 분기 예측기의 복잡도를 차등 배치하는 기법이 매우 효율적임을 의미한다.

참고 문헌

- [1] Ayose Falcon and Jared Stark, "Prophet/Critic Hybrid Branch Predictor," In Proceedings of the 31st Annual International Symposium on Computer Architecture, pp.250-262, Munchen, Germany, June, 2004
- [2] M. Evers, P.-Y. Chang, and Y. N. Patt, "Using hybrid branch predictors to improve branch prediction accuracy in the presence of context switches," In Proceedings of the 23rd Annual International Symposium on Computer Architecture, pp.3-11, Philadelphia, PA, USA, May, 1996
- [3] E. Jacobsen, E. Rotenberg, and J. E. Smith, "Assigning confidence to conditional branch predictions," In Proceedings of the 29th Annual International Symposium on Microarchitecture, pp.142-152, Paris, France, Dec., 1996.
- [4] D. A. Jimenez, "Fast path-based neural branch prediction," In Proceedings of the 36th Annual International Symposium on Microarchitecture, pp.243-252, San Diego, CA, USA, Dec., 2003.
- [5] G. H. Loh and D. S. Henry, "Predicting conditional branches with fusion-based hybrid predictors," In Proceedings of the International Conference on Parallel Architecture and Compilation Techniques, pp.165-176, Charlottesville, VA, USA, Sept., 2002.
- [6] Daniel A. Jimenez, Calvin Lin, "Dynamic Branch Prediction with Perceptron," In Proceedings of the International Symposium on High Performance Computer Architecture, pp.197-206, Nuevo Leone, Mexico, Jan., 2001.
- [7] C. Torregiani, H. Oprins, B. Vandeveld, E. Beyen, I. De Wolf, "Compact Thermal Modeling of Hot spots in Advanced 3D-Stacked ICs," In Proceedings of Electronics Packaging Technology Conference, pp.131-136, Singapore, Dec., 2009.
- [8] A. K. Coskun, J. L. Ayala, D. Atienza, T. S. Rosing, and Y. Leblebici, "Dynamic Thermal Management in 3D Multicore Architectures," In Proceedings of Design, Automation & Test in Europe Conference & Exhibition, pp.1410-1415, Nice, France, Apr., 2009.
- [9] Joyner J.W, Zarkesh-Ha. P, Meindl J.D, "A Stochastic Global Net-length Distribution for a Three-Dimensional System on Chip (3D-SoC)," In Proceedings of the 14th IEEE International ASIC/SOC Conference, pp.147-151, Arlington, VA, USA, Sep., 2001.
- [10] K. Puttaswamy, G. H. Loh, "Thermal Analysis of a 3D Die Stacked High Performance Microprocessor," In Proceedings

of ACM Great Lakes Symposium on VLSI, pp.19-24, Philadelphia, USA, 2006.

[11] Y. Xie, et al., "Design Space Exploration for 3D Architecture," ACM Journal of Emerging Technologies for Computer Systems, Vol.2. No.2, pp.65-103, April, 2006.

[12] J. Donald and M. Martonosi, "Techniques for multicore thermal management: Classification and new exploration," In Proceedings of 33rd International Symposium on Computer Architecture, pp.78-88, Boston, MA, USA, June, 2006.

[13] A. Baniasadi and A. Moshovos, "Instruction flow-based front-end throttling for power-aware high-performance processors," In Proceedings of the International Symposium on Low power Electronics and Design, California, USA, Aug., 2001.

[14] J. H. Choi, J. H. Kong, E. Y. Chung and S. W. Chung, "A Dual Integer Register File Structure for Temperature-Aware Microprocessor," Journal of KIISE: Computer System and Theory, Vol.35, No.12, pp.540-551, 2008.

[15] Chang Joo Lee, Hyesoon Kim, Onur Mutlu, Yale N. Patt, "Performance-Aware Speculation Control using Wrong Path Usefulness Prediction," In Proceedings of the International Symposium on High Performance Computer Architecture, pp.39-49, Salt Lake City, UT, USA, Feb., 2008.

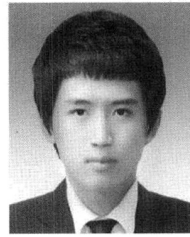
[16] A. K. Coskun, A. B. Kahng, T. S. Rosing, "Temperature- and Cost-Aware Design of 3D Multiprocessor Architectures," In Proceedings of 12th Euromicro Conference on Digital System Design and Architectures, Methods and Tools, pp.183-190, Patras, Greece, Aug., 2009.

[17] SPEC CPU2000 Benchmarks, available at <http://www.specbench.org>

[18] D. C. Burger, and T. M. Austin, "The SimpleScalar tool set, version 2.0," ACM SIGARCH CAN, Vol.25, No.3, pp.13-25, Jun., 1997.

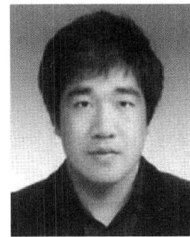
[19] S. McFarling, "Combining branch predictors," Technical Report TN-36, Compaq Western Research Lab., June, 1993.

[20] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-Aware Microarchitecture," In Proceedings of the 30th International Symposium on Computer Architecture, pp.2-13, San Diego, CA, USA, June, 2003.



안진우

e-mail : ajw0411@gmail.com
 2010년 전남대학교 전자컴퓨터공학과 (공학사)
 2012년 전남대학교 전자컴퓨터공학과 석사
 관심분야: SoC 설계, 임베디드 하드웨어 설계, 컴퓨터구조



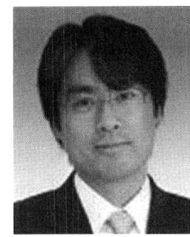
최홍준

e-mail : chj6083@gmail.com
 2009년 전남대학교 전자컴퓨터공학부 (공학사)
 2011년 전남대학교 전자컴퓨터공학과(석사)
 2011년~현 재 전남대학교 전자컴퓨터공학과 박사과정
 관심분야: 저전력 설계, 고성능 컴퓨팅, 병렬처리, 컴퓨터 구조



김종면

e-mail : jmkim07@ulsan.ac.kr
 1995년 명지대학교 전기공학사
 2000년 University of Florida ECE(석사)
 2005년 Georgia Institute of Technology ECE(박사)
 2005년~2007년 삼성종합기술원 전임연구원
 2007년~현 재 울산대학교 전기공학부 교수
 관심분야: 임베디드 SoC, 컴퓨터구조, 프로세서 설계, 병렬처리



김철홍

e-mail : chkim22@chonnam.ac.kr
 1998년 서울대학교 컴퓨터공학사
 2000년 서울대학교 컴퓨터공학부(석사)
 2006년 서울대학교 전기컴퓨터공학부 (박사)
 2005년~2007년 삼성전자 반도체총괄 SYS.LSI사업부 책임연구원
 2007년~현 재 전남대학교 전자컴퓨터공학부 교수
 관심분야: 임베디드시스템, 컴퓨터구조, SoC설계, 저전력 설계