

PARSEC을 이용한 TILE-Gx36 다중코어 프로세서의 성능 평가 및 분석

이보선[†] · 김한이[†] · 유현창^{††} · 서태원^{†††}

요 약

본 논문은 다중코어의 성능을 평가하고 분석하기 위해 TILE-Gx36(Gx36) 다중코어 프로세서를 사례로 연구하였다. Gx36의 성능 평가는 비교적 최신 병렬 벤치마크인 PARSEC을 이용하였고, 성능 분석을 돕기 위한 비교 시스템으로 인텔의 Core i7 (i7)과 Atom을 사용하였다. 실험결과 2의 제곱으로 동시에 수행 가능한 스레드를 발생시켰을 때, Gx36은 i7보다 평균 2.73배 낮은 성능을 보였으며, Atom보다는 평균 1.93배 높은 성능을 보였다. Gx36은 비교 프로세서보다 상대적으로 큰 Last-Level Cache(LLC)를 갖고 있음에도 불구하고, 가장 많은 LLC miss를 발생시켰다. 이는 Gx36이 기대치 이하의 성능을 보이는 주된 이유로 판단되며, DDC가 일반적 고성능 컴퓨팅을 위한 캐시구조로 적절하지 않음을 보여준다. 다중코어 시스템의 실적을 통한 성능평가는 향후 다중코어 구조개선 및 올바른 방향 설정을 위한 객관적인 자료를 제공한다.

주제어 : TILE-Gx36, 다중코어, 성능분석, PARSEC, Core i7, Atom

Performance evaluation and analysis of TILE-Gx36 many-core processor with PARSEC benchmark

Boseon Lee[†] · Han-Yee Kim[†] · Heonchang Yu^{††} · Taeweon Suh^{†††}

ABSTRACT

This paper evaluates and analyzes the performance of TILE-Gx36(Gx36), a many-core processor. The PARSEC parallel benchmark suite was used to measure the performance, and Core i7 (i7) and Atom are used for the performance comparison. When experimented with the maximum number of threads that can be executed concurrently on each machine, Gx36 showed a 2.73× inferior performance to Core i7 and a 1.93× superior performance to Atom. Gx36 has the largest Last Level Cache(LLC) among the compared processors. Nevertheless, it reported the biggest number of LLC misses, which, we strongly believe, is the major culprit for lower performance than expected. Our study suggests that the DDC employed in Gx36 is not a favorable cache structure for the general-purpose high-performance computing. The actual measurement with off-the-shelf machine provides non-biased data for polishing the future many-core architecture.

Keywords : TILE-Gx36, Many-core, Performance Evaluation, PARSEC, Core i7, Atom

[†] 정 회 원: 고려대학교 컴퓨터교육과 석박통합과정
^{†††} 종신회원: 고려대학교 컴퓨터교육과 부교수(교신저자)

^{††} 종신회원: 고려대학교 컴퓨터교육과 정교수

논문접수: 2013년 12월 20일, 심사완료: 2014년 1월 12일, 게재확정: 2014년 01월 23일

* 본 논문은 2013년 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행되었음 (NRF-2012R1A1A2008231)

1. 서론

2000년대 초반까지는 컴퓨터의 성능 향상을 위해 주로 싱글코어의 구조 개선에 집중하였다. 즉, 단일 코어의 파이프라인 단계를 증가시키고 이에 따른 크리티컬 패스(critical path)의 지연시간 감소를 통해 동작 주파수를 높여 성능을 향상 시켰다. 하지만, 이 방법은 높은 발열에 따른 파워벽(Power Wall)[1]에 직면하였고, 이후 산업계에서는 비교적 간단한 코어 여러 개를 하나의 칩에 집적하는 멀티코어(Multi-core)를 연구 개발 하였다.

멀티코어는 비교적 간단한 코어들에 여러 개의 스레드(thread)를 동시에 수행함으로써 컴퓨터의 성능 향상을 도모한다. 현재 멀티코어 기술은 데스크톱은 물론 전력에 민감한 태블릿 PC 및 스마트폰 등 휴대기기에까지 사용되고 있다. 최근에는 8 코어 기반의 태블릿 PC 및 스마트폰 용 프로세서(Exynos)[2]까지 출시되었다. 멀티코어의 개발 이후, 반도체 기술의 발전과 함께 하나의 칩에 수십 또는 수백 개의 코어를 집적 할 수 있는 다중코어(Many-core) 프로세서가 연구 개발되고 있다. <표 1>은 현재 출시된 다중코어 프로세서들을 보여준다.

코어의 개수 외에 멀티코어와 다중코어를 구분하는 주요한 차이점은 캐시 구조와 코어들을 연결하는 방식이라고 할 수 있다. 멀티코어는 물리적으로 하나인 Last-Level Cache(LLC)를 모든 코어들이 공유하는 반면[7][8], 다중코어는 다수의 코어에서 발생하는 요청을 동시에 효과적으로 처리하기 위해 LLC를 물리적으로 분산하여 설계하는 경향이 있다[3][4]. 또한, 멀티코어는 공유버스나 링(Ring)과 같은 비교적 간단한 연결망을 주로 사용한다[7][8]. 하지만, 다중코어는 다수의 코어에

서 발생하는 메모리 접근 요청이 동시에 분산 캐시에 전달될 수 있도록, 확장성을 제공하는 그물망(Mesh)과 같은 구조를 사용한다[4].

다중코어 및 컴퓨터 시스템에 대한 성능 분석은 객관적인 데이터에 근거하여 향후 실질적 개선 방향을 제시할 수 있다. 하지만, 현재까지 다중코어 시스템의 성능에 대한 실험 및 분석 연구 결과는 극히 제한적이다. 본 논문에서는 실제 다중코어의 성능을 평가하기 위해 Tiler사의 TILE-Gx36(Gx36)을 사용하여 사례 연구를 하였다. Gx36은 36개의 코어를 갖는 다중코어로 네트워크와 멀티미디어 처리에 적합한 프로세서로 소개되고 있지만, 내부는 일반적인 다중코어의 구조와 크게 다르지 않다. 따라서 범용 벤치마크를 이용한 Gx36의 성능평가 및 분석은 다중코어에 대한 실증적인 데이터를 제공할 수 있다는 면에서 의미가 있다.

본 논문에서는 Gx36의 성능 분석을 돕기 위한 비교 프로세서로 멀티코어 기반의 인텔 Core i7(i7)과 인텔 Atom D2700(Atom)을 사용하였다. i7과 Atom은 각각 4개, 2개의 물리적인 코어를 갖으며 하이퍼스레딩(Hyper-threading(HT))기술을 적용하여 각각 8개, 4개의 스레드를 동시에 수행할 수 있는 프로세서이다. 성능분석을 위해 PARSEC 벤치마크[9]를 이용하였고, 스레드 증가에 따른 실행시간 및 캐시 성능을 측정하였다.

논문의 구성은 다음과 같다. 2장에서 관련연구에 대해 조사하였고, 3장에서는 실험에 사용된 프로세서의 특성에 대해 자세히 서술한다. 4장에서는 PARSEC 벤치마크의 특성을 요약하였고, 일부 workload의 실행 시 제약사항과 실험 환경을 기술하였다. 5장과 6장은 본 연구의 실험결과와 분석 그리고 결론을 내린다.

<표 1> 다중코어 프로세서 모델

모델명	Intel Xeon Phi 7120 [3]	TILE-Gx72 [4]	TILEPro64 [4]	SPARC T5 [5]	Tesla K20X [6]
코어의 수	61 코어 (코어 당 4 스레드 = 244 스레드)	72 코어	64 코어	16 코어 (코어 당 8 스레드 = 128 스레드)	2688 CUDA 코어
클럭 속도	1.238 GHz	1.2 GHz	866 MHz	3.6 GHz	732MHz
코어연결방식	Ring	Mesh	Mesh	Fully connected	-
LLC	30.5 MB	18 MB	5.6 MB	8 MB	L1 : 64KB L2 : 1.5MB
출시년도	2013년	2012년	2011년	2013년	2012년

2. 관련연구

Bienia는 [9] 에서 SPLASH-2와 같은 기존 벤치마크들의 문제점을 개선한 PARSEC을 발표하였다. 또한, Bienia는 [10]에서 PARSEC과 SPLASH-2의 특성을 비교하였다. Bienia가 언급한 기존 벤치마크의 문제점은 아래와 같은 3가지로 구분된다.

첫째, 벤치마크의 노후이다. 멀티코어 성능평가에 많이 사용되고 있는 SPLASH-2는 1990년대에 개발된 벤치마크로 빅데이터 처리와 같은 최신의 workload를 지원하지 않는다. 또한 파이프라인 모델과 같은 병렬 모델도 지원하지 않는다.

둘째, 특정 어플리케이션에 편향된 벤치마크이다. SPEC2006은 병렬 머신을 지원하지 않고, 멀티코어를 위한 벤치마크들은 고성능 컴퓨팅 어플리케이션에 초점이 맞추어져 있었다.

셋째, SPLASH-2 나 SPEC과 같은 주요 벤치마크를 제외한 ALPBench, BioParallel, MediaBench, MineBench 와 같은 벤치마크들은 특정 목적에 초점이 맞추어져 범용적으로 사용하기에는 무리가 있다.

Bhadoria[11]는 Athlon, Atom, Conroe, NiagaraII, Pentium4, Phenom, Xeon 등의 프로세

서를 사용하여 PARSEC 벤치마크의 특성을 연구하였다. 실험에서는 캐시 성능과 주기억장치의 속도, 그리고 스레드 확장성에 따른 결과를 분석하였다. 실험 결과 과반수의 어플리케이션은 8개의 스레드까지 스레드의 증가에 따라 수행시간이 선형적으로 감소하였다. 또한, bodytrack, vips, x264의 수행 속도는 메모리 채널의 숫자에 따라 민감한 차이를 보였고 canneal과 streamcluster는 메모리 속도에 따라 민감한 수행시간의 차이를 보였다.

Kim[12] 은 Tiler사의 TilePro64과 Intel의 i7 및 Atom을 대상으로 SPLASH-2와 CINT2006를 사용해 성능평가를 하였다. 실험결과, TilePro64는 i7에 비해 19.2배 낮은 성능을 나타냈고, Atom에 비해 2.6배 낮은 성능을 보였다.

본 논문은 최근에 출시된 Gx36 다중코어 프로세서를 비교적 최신의 병렬 벤치마크인 PARSEC을 이용하여 성능을 평가하고 분석했다는 점에서 Bhadoria와 Kim의 연구와 차이가 있다.

3. Gx36 다중코어 및 비교 프로세서

실험을 위한 다중코어 프로세서인 Gx36과 성능 비교를 위한 프로세서인 i7과 Atom의 특성은 <표 2>와 같다.

<표 2> 실험에 사용된 프로세서의 하드웨어 특성 및 시스템 환경

프로세서	Tilera		Intel	
		Tile-Gx36 [4]	Core i7 950 [8]	Atom D2700 [13]
물리적 코어 수		36	4	2
논리적 코어 수		36	8 with HT	4 with HT
CPU 구조		64-bit 3-way VLIW	64-bit 4-way Superscalar	64-bit 2-way Superscalar
코어 동작속도		1.20 GHz	3.07 GHz	2.13 GHz
캐시	L1I	32KB (2-way)	32KB (4-way)	32KB (8-way)
	L1D	32KB (2-way)	32KB (8-way)	24KB (6-way)
	L2	256KB (8-way)	256KB (8-way)	1MB (8-way)
	L3	9MB - 256KB (DDC)	8MB (16-way)	-
캐시 블록 사이즈		64 bytes	64 bytes	64 bytes
LLC 일관성 규약		Directory-based	4-bit vector for internal cores, MESIF for ccNUMA architecture	MESI
주기억장치 (실제 운영체제에서 인식하는 크기)		16GB DDR3 (15.85GB)	4GB DDR3 (3.7GB)	4GB DDR3 (3.7GB)
운영체제		Linux 2.6.40.38	Cent OS 6.1 (2.6.32-131.21)	Cent OS 6.3 (2.6.32-279)
출시년도		2012년	2008년	2011년

3.1 Tile-Gx36

Tilera사의 Gx36은 <그림 1>과 같이 36개의 노드가 6 x 6 그물망(mesh) 구조로 연결된 다중 코어 프로세서이다. 각 노드는 CPU코어와 L1, L2 캐시, 그리고 스위치로 구성되어 있다. 코어는 3개의 명령어를 동시에 처리하는 Very Long Instruction Word(VLIW)구조를 가지고 있고, 그물망 연결을 통해 다른 노드의 L2 캐시에 접근할 수 있다. Tilera는 이를 Dynamic Distributed Cache(DDC)라 명명하였다. 각 노드의 코어는 DDC를 통해 자신의 L2캐시를 제외한 35개의 L2 캐시가 마치 L3 캐시와 같은 역할을 하는 것으로 인식한다. 이를 통해 각 코어에서의 주기억장치 접근을 감소시킬 수 있다. Tilera사에 의하면 Gx36은 병렬화하기 적합한 네트워크와 멀티미디어 어플리케이션에 최적화 되어 있다.

3.2 Core i7 950 과 Atom D2700

i7과 Atom은 각각 Nehalem 마이크로아키텍처와 Bonnell 마이크로아키텍처를 기반으로 설계된 프로세서이다. i7은 하나의 다이에 4개의 물리적인 코어가 집적되어 있고 HT기술을 통해 총 8개의 스레드를 동시에 수행 할 수 있다. Atom은 2개의 물리적인 코어를 갖고 HT를 통해 4개의 스레드를 동시에 수행 할 수 있다. i7은 세 단계의 캐시(L1, L2, L3)를 가지고 있고, Atom은 두 단계의 캐시(L1, L2)를 갖는다.

<표 3> PARSEC workload 특성

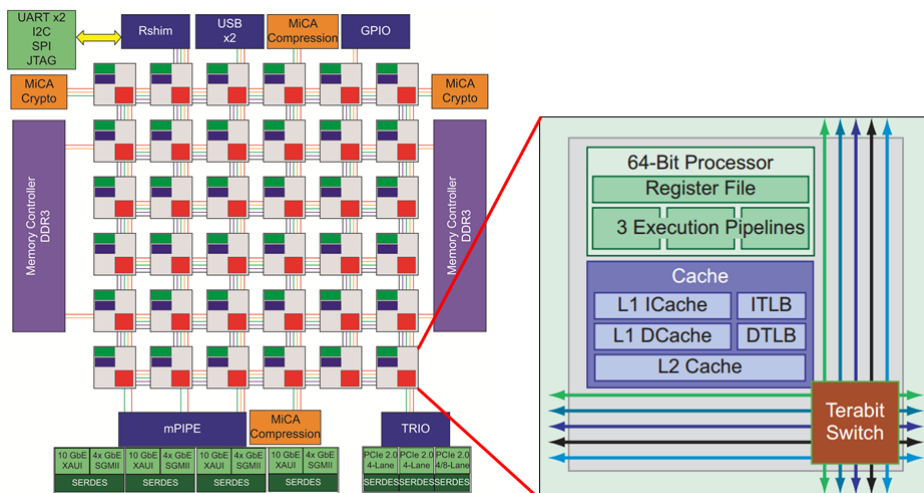
blackscholes	calculates portfolio price using Black-Scholes PDE
facesim	physics simulation, models a human face
fluidanimate	physics simulation, animation of fluids
freqmine	data mining application
streamcluster	kernel to solve the online clustering problem
swaptions	computes portfolio prices using Monte-Carlo simulation

4. 실험환경

4.1 PARSEC 벤치마크

PARSEC은 최신 어플리케이션의 경향을 반영하고 다양한 workload를 지원하는 병렬 벤치마크이다. 또한 테스트 및 개발을 위한 Input Set과 시뮬레이션을 위한 Input Set등 다양한 크기의 Input Set을 제공한다. 실험에서는 native input set을 사용하였다.

<표 3>은 본 논문에서 실험한 PARSEC workload의 특성을 보여준다. blackscholes과 swaptions는 재무분석, fluidanimate와 facesim은 애니메이션, 그리고 freqmine은 데이터 마이닝 관련 어플리케이션이다.



<그림 1> Gx36의 하드웨어 구조 및 각 코어의 구조 [4]

4.2 PARSEC Workload 수행 제약사항

PARSEC은 총 12개의 workload를 제공하고 있다. 하지만 본 실험에서는 아래의 제한으로 <표 3>에 나열된 workload를 수행하였다.

첫 번째, 라이브러리 제한이다. workload를 수행하려면 각 어플리케이션을 컴파일하기 위한 라이브러리가 필요하다. 하지만 Gx36 리눅스 시스템은 PARSEC 수행 시 필요한 라이브러리 중 일부를 제공하지 않아 모든 workload에 대한 실험에는 어려움이 있었다.

두 번째, 주기억장치 크기의 제한이다. dedup의 경우에는 최소로 요구되는 주기억장치의 크기가 4GB이다. 하지만, i7과 Atom의 경우 운영체제에서 실제 인식하는 주기억장치의 크기가 4GB 미만이기 때문에 dedup을 수행하는데 제약이 따랐다.

5. 실험 결과 및 분석

5.1절에서는 각 프로세서 별 스레드 수에 따른 실행시간 및 speedup을 측정하고 비교하였고, 5.2절에서는 LLC 접근과 LLC miss를 통해 LLC가 성능에 미치는 영향을 비교하고 분석하였다.

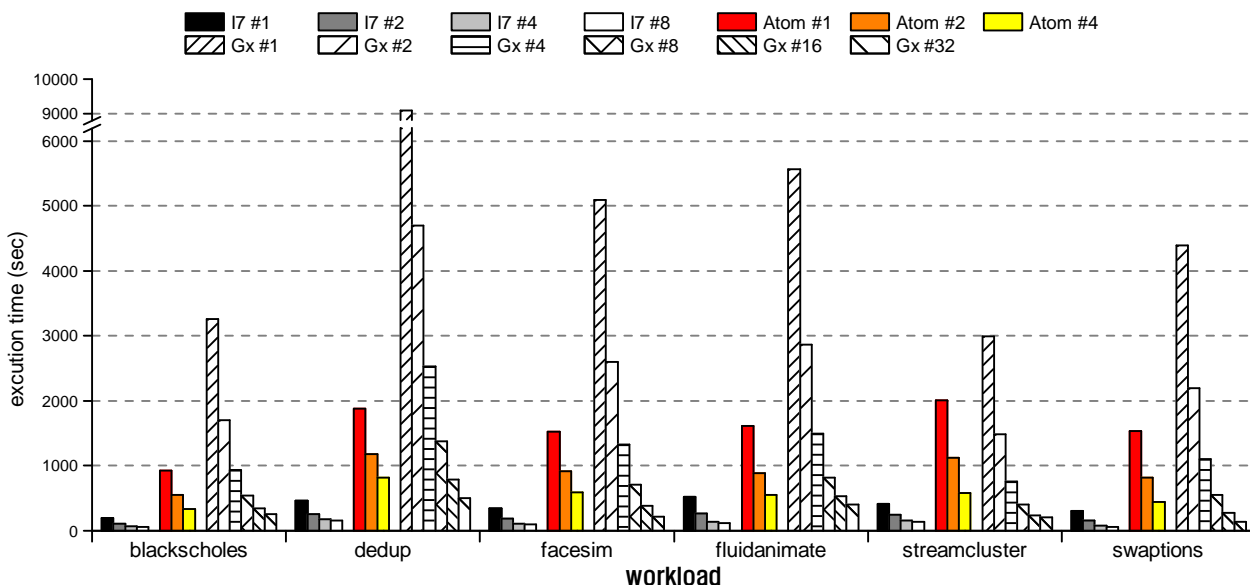
5.1 스레드 수에 따른 실행시간 및 speedup

<그림 2>는 스레드 수 증가에 따른 workload

의 수행 시간을 측정된 그래프이다. <그림 2>의 범례에서 #number는 스레드의 수를 표현한다. 예를 들면, Gx #32는 Gx36에서 workload를 32개의 스레드로 수행했을 때를 의미한다. 실험은 스레드 1개부터 시작하여 2의 제곱으로 각 프로세서가 동시에 최대 수행할 수 있는 스레드까지 증가시키며 반복하였다. 즉, Gx36에서는 최대 32개, i7은 최대 8개, Atom은 최대 4개의 스레드까지 증가시켰다.

실행시간 측정 결과, 하나의 스레드로 실행했을 때 평균적으로 Gx36은 i7보다 13.5배 느렸고, Atom에 비해서도 3.2배 느렸다. 즉, 싱글코어의 성능은 Gx36이 i7과 Atom에 비해 현저히 저조하다는 것을 보여준다. 그리고 동시에 수행할 수 있는 최대의 스레드로 실행 했을 때, Gx36은 i7보다 평균 2.73배 낮은 성능을 나타냈으며, Atom에 비해서는 평균 1.93배 높은 성능을 보였다. 즉, Gx36에서 스레드의 수를 32개로 증가시켰을 때 Atom에서 4개의 스레드로 수행한 것에 비해 약 2배의 성능우위만을 보여주었다.

<그림 3>은 스레드 1개의 성능을 기준으로 스레드 증가에 따른 speedup 그래프이다. swaptions은 Gx36에서 32 스레드로 수행 시 성능이 31.9배로 거의 선형적으로 향상된 반면, blackscholes은 32 스레드로 수행 시 12.9배로 확장성 (scalability)가 가장 좋지 않았다. i7의 경우도



<그림 2> i7, Atom, Gx36에서 스레드의 수에 따른 workload 수행 시간

speedup이 가장 좋은 workload는 swaptions으로 8 스레드로 수행 시 성능이 4.9배 향상 되었다. 반면, facesim은 8 스레드로 수행 시 성능 향상은 2.9배로 가장 저조한 향상율을 나타냈다. Atom도 i7과 유사한 경향을 보였다. 즉, swaptions을 4 스레드로 수행했을 때 3.47배로 좋은 확장성을 보여 주었고, facesim은 4 스레드로 수행 시 2.4배의 향상율을 보여주었다.

[9]에 따르면 swaptions은 메모리 배리어(memory barrier)와 스레드 간의 동기화를 위한 lock이 현저히 적다. 이는 병렬성이 좋다는 것을 의미하며 실험에서도 세 프로세서에서 모두 speedup 효율이 가장 좋았다. 특히, Gx36에서는 스레드 수에 따라 거의 선형적인 성능 증가를 보여주었다.

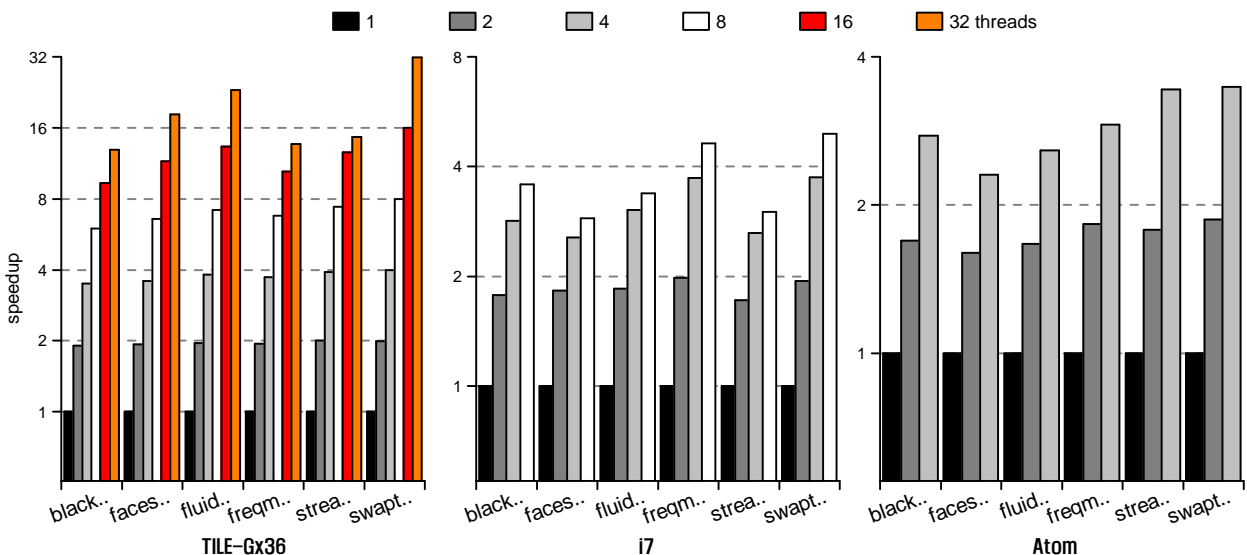
<그림 3>에서 또한 주목할 것은 i7과 Atom의 경우 스레드가 최대로 증가할 때 speedup이 비교적 낮다는 것이다. 즉, i7은 스레드가 4개에서 8개로 증가할 때, 그리고 Atom은 2개에서 4개로 증가할 때 speedup 향상율이 크지 않다. 이는 운영체제 작업을 위해 시스템에서 수행하는 스레드로 인해 전체 시스템이 수행하는 스레드의 수가 논리코어의 수를 초과하기 때문으로 판단된다. 즉, 운영체제 스레드들은 PARSEC 스레드들과 스워칭하며 코어를 점유하여 수행하기 때문에, PARSEC 스레드들 간 수행시간에 불균형이 생길 수 있다. 또한, 스레드 간의 동기화를 거치는 과

정에서 busy-waiting하는 시간이 많아져 전체적인 성능의 저하가 발생하는 것으로 판단된다.

5.2 LLC 성능

컴퓨터에서 메모리 계층구조는 컴퓨터 성능에 막대한 영향을 미친다. <표 2>에 정리한 것과 같이 Gx36은 DDC구조를 이용하여 LLC역할을 하는 9MB에 가까운 캐시가 있다. 이에 비해 i7은 8MB, Atom은 1MB의 LLC를 가지고 있다. Gx36이 i7과 Atom에 비해 상대적으로 큰 캐시를 갖고 있음에도 성능이 낮은 이유를 분석하기 위해 LLC의 성능을 측정 비교하였다. 데이터 수집은 OProfile[14]을 사용하여 하였다.

<그림 4>는 스레드 수에 따른 각 프로세서의 LLC 접근과 LLC miss를 나타낸 그래프이다. 여기서 y축은 로그스케일(log scale)이다. 실험 결과, Gx36은 평균적으로 약 940,000번, i7은 약 28,500번, Atom은 약 110,000번의 LLC 참조가 발생했다. 즉, Gx36은 i7에 비해 33배, Atom에 비해서는 8.5배 만큼 LLC접근이 많았다. 또한, LLC miss는 Gx36이 약 130,000번, i7은 약 24,700번, Atom은 약 36,000번이 발생했다. 즉, Gx36은 i7에 비해 6.5배, Atom에 비해서는 3.6배 miss발생 횟수가 많았다. LLC miss가 발생하면 주기억장치에 접근해야 하며, 긴 주기억장치 접근 시간으로 인해 벤치마크 수행시간이 길어지게 된다.



<그림 3> Gx36, i7, Atom의 스레드 수에 따른 speedup

Gx36의 LLC구조를 분석하면 상대적으로 많은 LLC접근과 miss에 대한 해석이 가능하다. <그림 1>과 같이 Gx36의 각 노드는 Local L2캐시를 가지고 있고, 다른 노드의 L2캐시들을 L3캐시처럼 사용하는 구조로 되어있다. 따라서 하나의 노드에서만 프로그램이 수행될 때에는 다른 노드의 L2 캐시들을 독점적으로 사용할 수 있다. 하지만 스레드의 수가 증가하면 L2캐시들을 스레드들(즉, 코어들) 간에 공유하게 되어 각각의 코어가 실제로 사용할 수 있는 L3캐시의 크기는 점점 줄어들게 된다. 최악의 경우 Gx36에 있는 모든 코어들이 사용될 때에는 스퀴싱(thrashing)이 발생할 수 있고, 이로 인해 LLC 효과가 감소 될 수 있다. 따라서 물리적인 LLC 크기는 9MB로 가장 크지만, 실제 효과적인 크기는 작다고 볼 수 있다. 이는 DDC가 고성능 컴퓨팅에서 전통적인 병렬 벤치마크와 같은 workload의 효율적인 수행을 위한 다중코어 프로세서의 캐시 구조로 적절하지 않음을 시사한다.

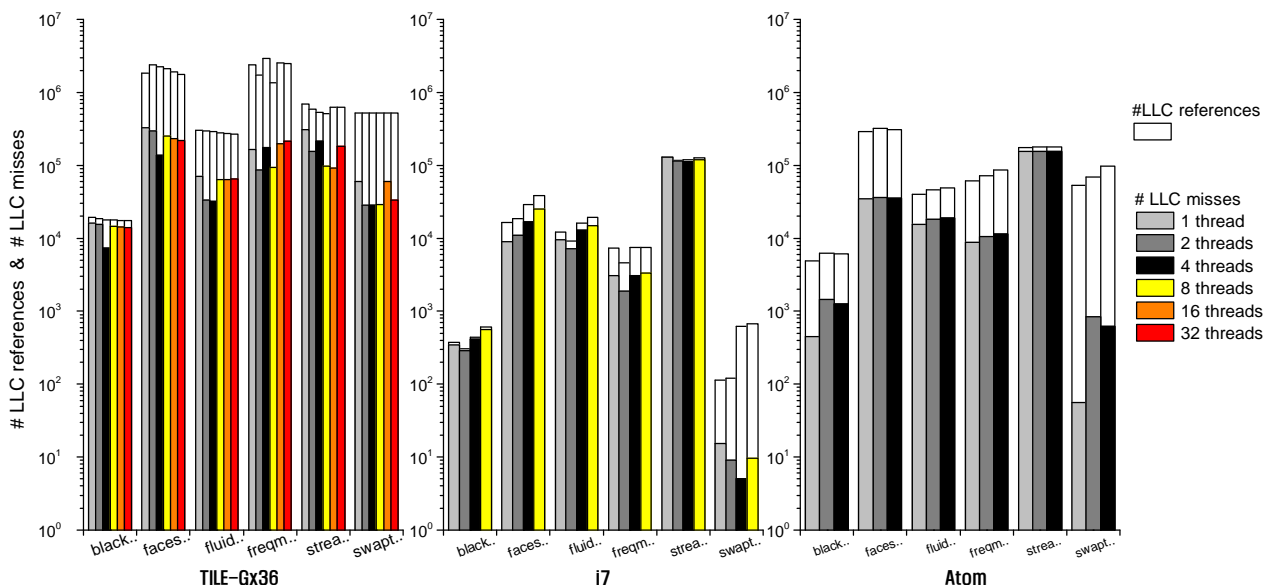
5.3 개선방안 제안

Gx36은 i7과 Atom의 캐시구조와 달리 L2 캐시와 LLC가 실질적으로 같은 레벨에 존재한다. 이로 인해 Gx36의 캐시 성능은 크기에 비해 효율성이 떨어졌다. Gx36의 고성능 컴퓨팅 시장 진입을 위해서는 캐시 구조의 변화가 필수적이라고 판단

된다. Gx36의 성능 개선을 위해서는 L1, L2는 각 코어가 전용하는 캐시로 유지하고, 3D 구조를 이용하여 LLC 공유 캐시를 설계하는 것이 하나의 개선방안이 될 것으로 판단된다.

6. 결론

본 논문은 Gx36를 사례로 다중코어 프로세서의 성능을 평가 및 분석하였다. 성능 평가를 위한 벤치마크로는 PARSEC을 사용하였고 성능비교를 위해 멀티코어 프로세서인 i7과 Atom을 사용하였다. 실행시간 측정 결과, Gx36은 i7보다 평균 2.73배 낮은 성능을 보였고, Atom에 비해서는 평균 1.93배 높은 성능을 보였다. Gx36은 i7과 Atom에 비해 큰 LLC를 갖고 있음에도 불구하고 많은 LLC miss가 발생했고, 이에 따른 잦은 memory 접근에 의해 성능저하가 발생했다. 이는 Gx36의 DDC가 일반적 고성능 컴퓨팅을 위한 캐시 구조로 적절하지 않음을 보여준다. 다중코어 시스템의 실측을 통한 성능평가는 향후 다중코어 구조개선 및 올바른 방향 설정을 위한 객관적인 자료를 제공한다.



<그림 4> 스레드의 수에 따른 LLC 접근과 LLC miss

참 고 문 헌

- [1] Power Wall, http://en.wikipedia.org/wiki/Multi-core_processor
- [2] Exynos, <http://en.wikipedia.org/wiki/Exynos>
- [3] Xeon Phi, <http://ark.intel.com/ko/products/family/71840>
- [4] Tilera Corporation, <http://www.tilera.com/>
- [5] Feehrer, J., Jairath, S., Loewenstein, P., Sivaramakrishnan, R., Smentek, D., Turullols, S., & Vahidsafa, A. (2013). *The Oracle Sparc T 5 16-core processor scales to eight sockets*. Micro, IEEE, 33(2), 48-57.
- [6] NVIDIA Corporation, <http://www.nvidia.com/>
- [7] Core 2 duo, <http://www.intel.com/products/processor/core2duo/>
- [8] Core i7, <http://ark.intel.com/products/37150>
- [9] Bienia, C., Kumar, S., Singh, J. P., & Li, K. (2008). *The PARSEC benchmark suite: characterization and architectural implications*. In Proceedings of the 17th international conference on Parallel architectures and compilation techniques (pp. 72-81). ACM.
- [10] Bienia, C., Kumar, S., & Li, K. (2008). *PARSEC vs. SPLASH-2: A quantitative comparison of two multithreaded benchmark suites on chip-multiprocessors*. In Workload Characterization, 2008. IISWC 2008. IEEE International Symposium on (pp. 47-56).
- [11] Bhadauria, M., Weaver, V. M., & McKee, S. A. (2009). *Understanding PARSEC performance on contemporary CMPs*. In Workload Characterization, 2009. IISWC 2009. IEEE International Symposium on (pp. 98-107).
- [12] Kim, H. Y., Kim, Y. H., Yu, H., & Suh, T. (2013). *Performance evaluation of many-core systems: case study with TILEPro64*. IET Computers & Digital Techniques, 7(4), 143-154.
- [13] Atom, <http://ark.intel.com/products/59683>
- [14] OProfile, <http://oprofile.sourceforge.net>

이 보 선



2011 전북대학교
전자공학과 (학사)
2011~현재 고려대학교
컴퓨터교육과
석·박사 통합과정

관심분야: 컴퓨터구조, 임베디드 시스템
E-Mail: l2bs@korea.ac.kr

김 한 이



2012 고려대학교
컴퓨터교육과 (학사)
2012~현재 고려대학교
컴퓨터교육과
석·박사 통합과정

관심분야: 컴퓨터구조, 임베디드 시스템
E-Mail: hanyeemy@korea.ac.kr

유 헌 창



1989 고려대학교 이과대학
전산학과 (이학사)
1991 고려대학교 대학원
전산학과 (이학석사)

1994 고려대학교 대학원 전산학과 (이학박사)
1998~현재 고려대학교 컴퓨터교육과 교수
2006~2010 한국컴퓨터교육학회 부회장
관심분야: 클라우드 컴퓨팅, 분산 시스템,
결함포용 시스템, u-learning
E-Mail: yuhc@korea.ac.kr



서 태 원

1993 고려대학교
전기공학과 (학사)
1995 서울대학교
전자공학과 (석사)

2006 Georgia Institute of Technology Computer
Engineering(공학박사)

1995~1998 LG종합기술원 주임연구원

1998~2001 하이닉스반도체 선임연구원

2004~2004 Intel Corporation, Research
Intern, CA, USA.

2005~2006 Intel Corporation,
Research Intern, OR, USA.

2007~2008 Intel Corporation,
Systems Engineer, OR, USA.

2008~현재 고려대학교 컴퓨터교육과 부교수

관심분야: 컴퓨터구조, 임베디드 시스템, 컴퓨터교육,
멀티프로세서

E-Mail: suhtw@korea.ac.kr