

Periodic Mapping을 통한 프로세서 레지스터 파일의 온도 관리

허인구*, 박상현*, 김용주*, 윤중희*, 이진용* 백윤홍*

*서울대학교 전기공학부

e-mail:{igheo, shpark, yjkim, jhyoon, jylee}@optimizer.snu.ac.kr

ypaek@snu.ac.kr

Periodic Mapping : Thermal Management for Processor Register File

In-Goo Heo*, Sang-Hyun Park*, Yong-joo Kim*, Jong-hee W. Yoon*,
Jin-Yong Lee*, Yun-Heung Paek*

*Dept of Electrical Engineering, Seoul University

요 약

공정이 미세화 될수록 프로세서 상에서의 thermal management는 점점 중요해지고 있다. 칩의 온도가 임계 온도를 넘어 손상되거나, 시스템이 불능이 되는 상황을 방지하기 위해 그 동안 많은 기법들이 소개되어왔다. 하지만 이러한 기법들은 시스템 전체를 끄거나 느려지게 함으로써 상당한 양의 성능 저하를 가져왔다. 이 논문에서는 프로세서의 가장 중요한 Hotspot인 Register File의 온도 관리를 위한 기법으로 Periodic Mapping을 제안하고, 이를 기존의 기법들과 비교해 본다.

1. Introduction

현재의 반도체 공정은 45nm까지 세밀해졌으며 2011년에는 22nm에 이를 것으로 예상된다. 이렇게 공정이 세밀해지고 집적도가 높아짐에 따라 power density 역시 급격히 증가하고 있는데, 수년 전에 출시된 Intel Pentium 4 processor가 이미 $34 W/cm^2$ 을 넘는 것으로 알려졌다. 이러한 높은 power density는 많은 열을 발생시켜 칩의 온도가 빠르게 상승하게 한다. 칩의 온도가 임계 온도 이상까지 넘어가게 되면 칩의 일부 혹은 시스템 전체가 손상되게 된다. 이를 막기 위해 물리적으로는 더 성능이 뛰어난 cooling system을 구축함으로써 칩의 온도를 조절할 수 있다. 하지만 이는 packaging cost를 높이게 되어 제품의 단가를 높이게 된다. 이런 이유로, packaging을 바꾸지 않고 architecture 수준에서 수행할 수 있는 여러 Thermal Management 기법들이 제안되었으나, 이들 대부분은 칩 전체를 shut down 시키거나 slow down 시킴으로 인해 필요 이상의 성능저하를 유발시켰다.

프로세서의 여러 functional unit 중 Hotspot이 주로 되는 곳은 Register File, ALU, Branch Predictor 등이다. 이 중 Register File은 작은 사이즈로 인해 상대적으로 power density가 높아 가장 중요한 Hotspot이다 [1]. 이 Register File의 온도를 관리하기 위해 여러 가지 기법들이 소개되었는데, 이중 [2]는 추가적인 Register File을 이용하여 온도를 관리하는 “Migration Computation”이라는 기법을 제안하였다. 하지만 이 기법 역시 Register File이 시스템 전

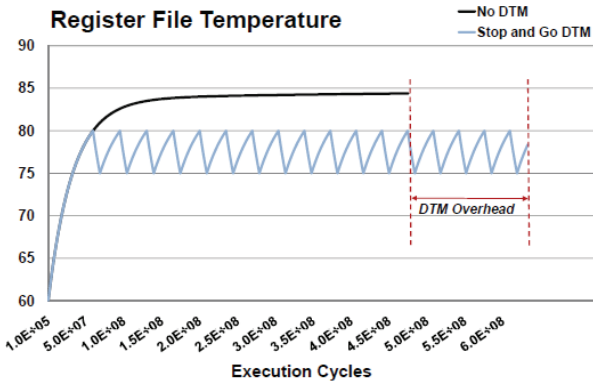
체의 bottleneck이 됨으로써 무시할 수 없는 성능 저하를 가져왔다.

이 논문에서는 [2]의 architecture를 이용한 Periodic Mapping 기법을 제안하고 이를 기존의 다른 기법들과 비교해 보도록 한다. 실험 결과, 평균적으로 전통적인 온도 관리 기법인 “Stop and Go”[3]와 “Migration Computation”이 각각 45.55%, 19.76%의 성능 저하를 가져온 데 비해, 제안된 “Periodic Mapping”은 10.95% 만의 성능 저하를 유발했다.

이 논문은 다음과 같이 구성된다. 2장에서는 기존의 기법들의 한계에 대해 알아보고, 이를 극복하기 위한 새로운 thermal management 기법의 필요성에 대해 이야기한다. 3장에서는 제안된 Periodic Mapping에 대하여 설명한다. 4, 5장은 실험에 대한 내용으로 실험 환경과 방법, 실험 결과로 구성되어 있으며, 6장에서는 결론을 맺도록 하겠다.

2. Motivation and Related Work

그림1은 SPEC95의 “Go” 벤치마크를 ALPHA 21364 프로세서에서 수행했을 때의 Register File 온도 그래프이다. 어떠한 Dynamic Thermal Management도 없을 때 Register File의 온도는 임계 온도로 설정된 80°C를 넘어가게 된다. 이러한 thermal violation을 방지하기 위해서 “Stop and Go” 기법은, Hotspot의 온도가 임계 온도에 도달 했을 때 프로세서 전체를 off 시키고, Hotspot의 온도

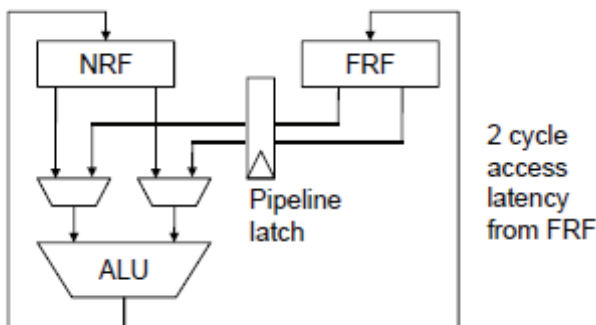


<그림 1> DTM의 적용 여부에 따른 온도 변화 비교

가 $T_{threshold}$ 까지 내려가는 것을 기다린다. 그림1은 이 $T_{threshold}$ 가 75°C일 때를 보여주고 있다. "Stop and Go" 기법은 안전한 온도 관리 기법이긴 하지만 그림1에서 보듯 33%라는 상당한 오버헤드를 발생시킨다. 이러한 성능 저하는 무시할 수 있는 수준이 아니다. "Go" 벤치마크보다 더욱 Register File의 power density를 높이는 벤치마크라면 33%라는 성능 저하 수치는 더욱 커질 수 있다.

이렇게 성능 저하가 상당한 것은 "Stop and Go"라는 기법 자체가 칩 전체를 shut down 하기 때문이다. 굳이 칩 전체의 동작을 중지시키지 않더라도 동작 전압과 frequency를 동적으로 낮춤으로써 칩의 온도를 낮추는 기법들이 꾸준히 각광받아 왔다. 이러한 DVS(Dynamic Voltage Scailing) 혹은 DVFS(Dynamic Volatage and Frequency Scailing) 기법들은 칩의 동작 속도를 늦추되 정지시키지 않음으로서 느리게나마 프로그램을 계속 수행할 수 있다는 장점을 가지고 있다.

하지만 이러한 DVS, DVFS 기법들 역시 특정 Hotspot으로 인해 칩 전체의 동작을 slow down 시켜 무시할 수 없는 만큼의 성능 저하를 가져온다. 예를 들어 Register File이 Hotspot인 경우 나머지 functional unit들의 온도는 임계 온도에 도달하지 않는 경우가 많지만, Register File의 온도가 떨어질 때까지 빠르게 동작할 수 없거나 중지되어야만 한다.



<그림 2> "Migration Computation" 기법을 위한 architecture
이러한 한계를 극복하기 위해 [3]에서는 추가적인

Register File을 포함한 프로세서 architecture를 제안하고 이를 활용하는 "Migration Computation" 기법을 소개하였다. 그림2와 그림3은 [3]에서 제안한 추가의 Register File을 나타내고 있다. "Migration Computation" 기법은 프로세서의 동작 중 NRF(Near Register File)의 온도가 임계 온도에 도달하면 4 cycle에 걸쳐 NRF의 값들을 FRF(Far Register File)로 옮기고 NRF 대신 FRF를 이용해 프로그램을 계속 수행한다. 하지만 그림3에서 보듯 FRF는 기존의 Integer Register File인 NRF에 비해 칩의 바깥 부분에 있으므로 1 cycle 만에 access가 되지 않는다. 따라서 FRF를 access할 때는 추가적인 1 cycle의 딜레이가 발생한다. FRF를 이용해 프로세서가 계속 동작을 하다가 다시 NRF의 온도가 $T_{threshold}$ 까지 내려가면 다시 FRF의 값을 NRF로 옮기고 NRF를 이용해 동작을 계속한다.

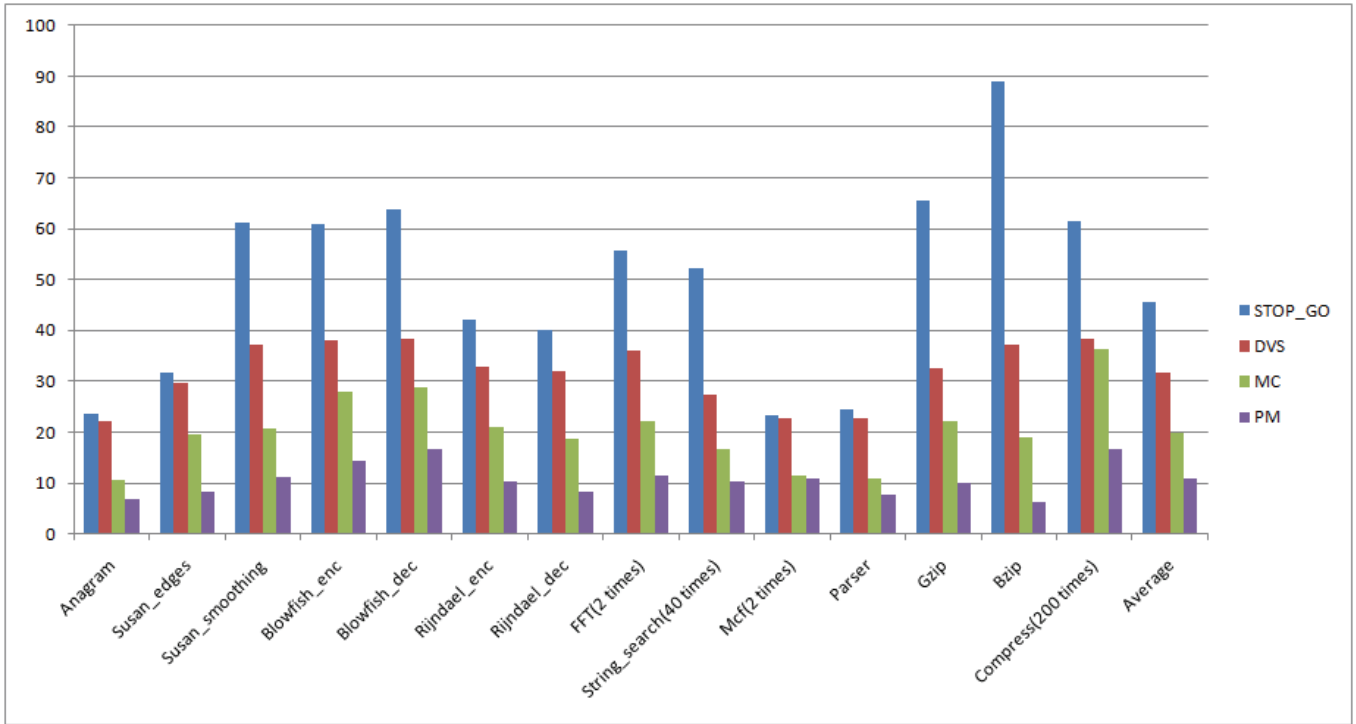
IntReg2 (FRF)	LdSt Q	Int Q	Int Q
FPMul	FPMap		IntReg1 (NRF)
FPReg			
FPAdd	FPQ	ITB	IntExec
B-Pred		DTB	
IL1		DL1	

<그림 3> FRF를 포함한 ALPHA 21364 프로세서의 FloorPlan

3. Periodic Mapping

"Migration Computation" 기법은 칩 전체를 shut down 하지 않는다는 장점이 있기는 하지만 FRF를 사용하여 수행하는 시간 동안에는 Register File이 bottleneck이 되어 성능 저하를 가져오게 된다. "Migration Computation" 기법은 NRF의 온도가 임계 온도에 도달할 때까지 어떠한 DTM도 적용하지 않는다. 이는 결과적으로 NRF의 온도가 임계 온도까지 올라가는 것을 방지하지 못한다. 이를 극복하기 위해 이 논문에서는 Periodic Mapping 기법을 제안한다.

Periodic Mapping은 주기적으로 instruction의 destination을 FRF에 할당하는 방식이다. 주기적으로 FRF에 값들을 할당하여 access하게 함으로써, NR를 access하는 횟수를 낮춰 온도가 좀 더 천천히 올라갈 수 있게 한다. 이 Periodic Mapping에서 가장 중요한 것은 이 주기를 얼마로 할 것인가인데, 이는 application의 특성에 따라 달라질 수 있다. 이를 위해 Application 별로 최적화된 주기를 찾아내고 그 주기에 따라 Register 할당을 한다. Register File의 온도가 임계 온도에 도달하지만 않는다면



<그림 4> 실험 결과 정리

임계온도 근처의 높은 수준으로 유지되는 것은 전혀 문제가 되지 않는다. 오히려 NRF의 온도가 높은 수준으로 유지된다는 것은 그만큼 FRF를 access하지 않는다는 의미이므로, FRF를 사용했을 때의 성능저하를 최소화한다는 뜻이 되기도 한다. 이를 바탕으로, application에 따라 NRF의 온도를 임계 온도 아래에서 최대로 유지할 수 있게 하는 주기를 찾고 이를 개별적으로 적용한다.

4. Experimental Environment

제안된 "Periodic Mapping" 기법을 다른 기법들과 비교하기 위해 [3]에서 사용된 실험 환경을 그대로 따르도록 하였다. [3]에서는 SimpleScalar[4]에 HotSpot[5]을 이식시킨 PTscalar[6]를 이용하여 cycle-accurate하게 프로세서의 functionality와 파워, 온도를 시뮬레이션 하였다. 이 논문의 실험 역시 PTscalar를 이용하였다. 원래의 PTscalar의 경우 130 nm 공정을 가정하여 만들어졌기 때문에 현재 많이 쓰이는 공정과는 차이가 있다. 공정이 세밀해 지면서 dynamic power만큼이나 leakage power에 대한 고려가 중요해지고 있는만큼 [7]을 참고하여 PTscalar의 파워 모델을 45nm 공정을 고려하여 수정하였다. 나머지 언급되지 않은 사항들은 PTscalar의 기본 설정들을 따랐음을 밝힌다.

5. Experimental Results

SPEC 2000과 Mibench의 벤치마크들 중 일부를 이용해 실험을 하였다. 벤치마크의 수행시간에 따라 온도 특성이 크게 달라지는 것을 보완하기 위해 모든 벤치마크는 약

400M~500M cycle 동안 수행하도록 하였다. 이것보다 긴 수행 cycle을 갖는 벤치마크는 반복적으로 수행하여 온도를 측정했다.

새로 제안된 "Periodic Mapping" 기법 이외에 "Stop and Go", "DVS", "Migration Computation" 기법들이 비교를 위해 같이 수행되었고, 어떠한 DTM도 적용하지 않았을 때를 Basecase로 하여 각각의 DTM들이 Basecase와 비교해 얼마만큼의 성능 저하를 가져오는 지 알아보았다. 실험 결과는 그림4, 표1과 같다.

기존의 DTM 기법들 중 칩 전체를 shut down하는 "Stop and Go" 기법은 4개의 DTM 기법 중 가장 많은 평균 45.55%의 성능저하를 보였다. 칩 전체를 slow down하는 "DVS" 기법 역시 평균 31.71%의 무시할 수 없는 성능

	STOP_GO	DVS	MC	PM
Anagram	23.57	22.20	10.56	6.98
Susan_edges	31.61	29.82	19.52	8.43
Susan_smoothing	61.09	37.32	20.70	11.11
Blowfish_enc	60.96	38.05	27.97	14.42
Blowfish_dec	63.82	38.34	28.85	16.65
Rijndael_enc	42.13	32.76	20.99	10.22
Rijndael_dec	40.17	31.86	18.63	8.33
FFT	55.59	35.96	22.18	11.35
String_search	52.22	27.37	16.56	10.38
Mcf	23.35	22.76	11.48	10.91
Parser	24.52	22.88	10.83	7.61
Gzip	65.60	32.55	22.11	9.97
Bzip	88.87	37.14	18.93	6.12
Compress	61.57	38.26	36.30	16.76
Average	45.55	31.71	19.77	10.94

<표 1> 기법들에 따른 성능 저하 비교(%)

저하를 보였다. "Migration Computation" 기법의 경우 칩 전체에 DTM을 적용하는 "Stop and Go"나 "DVS"에 비해 나아진 향상을 보였지만 여전히 20%에 가까운 평균 19.77%의 성능 저하를 보였다. 하지만 다른 기법들과 비교했을 때 "Migration Computation"은 칩의 일부에만 DTM을 적용한다는 점이 큰 장점으로 작용하여 "DVS"에 비해서도 10% 이상 성능 저하를 줄일 수 있었다. "Periodic Mapping"은 DTM을 임계 온도 도달 이전부터 적용하고, application을 분석하여 최적의 주기를 적용함으로써 평균적으로 단 10.94% 성능 감소만을 유발했다. 이는 "Migration Computation"과 같은 architecture와 packaging을 이용했다는 것을 감안할 때 굉장한 향상이라고 볼 수 있다. 벤치마크마다 약간의 차이가 있었지만 성능 저하는 최대 16.76%를 넘지 않았고, Bzip 벤치마크의 경우에는 단 6.12%의 성능 저하만이 있었을 뿐이었다.

6. Conclusions

프로세서의 가장 중요한 hotspot인 Register File의 온도를 관리하기 위한 기법으로 "Periodic Mapping"을 제안하였다. 칩 전체를 shut down하거나 slow down 시키는 기법들에 비해, hotspot에만 DTM을 적용함으로써 필요 이상의 성능 저하를 최소화할 수 있었고, 같은 architecture와 packaging을 사용한 "Migration Computation" 기법과 비교해서도 향상된 성능을 보여주었다. 이러한 "Periodic Mapping" 기법은 추가적인 functional unit을 이용한 기법 중의 하나로써 최근 이슈화 되고 있는 저온도 SoC 설계에 많이 이용될 수 있으리라 생각한다.

참고문헌

- [1] J. Deeney. Thermal modeling and measurement of large high power silicon devices with asymmetric power distribution. In *International Symposium on Microelectronics*, pages 300-305, 2002.
- [2] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan. Temperature-aware microarchitecture: Modeling and implementation. *ACM Trans. Archit. Code Optim.*, 1(1):94 - -125, 2004.
- [3] D. Brooks and M. Martonosi. Dynamic thermal management for high-performance microprocessors. In *HPCA*, pages 171 - -182, Washington, DC, USA, 2001. IEEE Computer Society.
- [4] <http://www.simplescalar.com>
- [5] <http://lava.cs.virginia.edu/HotSpot/>
- [6] <http://eda.ee.ucla.edu/PTscalar/>
- [7] D. Duarte, N. Vijaykrishnan, M. J. Irwin, and Y.-F. Tsai. Impact of technology scaling and packaging on dynamic voltage scaling techniques. In 15th Annual

IEEE International ASIC/SOC Conference, September 2002.

Acknowledgement

본 연구는 교육과학기술부/한국과학재단 우수연구센터 육성사업(과제번호 2009-0063249), 서울시 산학연 협력사업(10560), 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 국가지정연구실사업(No.2009-0083190)의 지원을 받아 수행되었습니다.