저전력을 위한 뱅크 선택 메커니즘과 새로운 동작 메커니즘을 이용한 직접사상 캐쉬 및 버퍼 시스템

이종성[O] 이정훈 김신덕
연세대학교 컴퓨터산업공학부
{dahui[O], ljh, sdkim}@cs.yonsei.ac.kr

A new direct-mapped cache with fully associative buffer for low power consumption by using bank-selection mechanism

Jongsung Lee[O] Junghoon Lee Shindug Kim
Department of Computer Science Yonsei University

본 논문은 서로 다른 두 구조의 캐쉬와 새로운 뱅크선별기를 이용하여, 보다 효율적인 뱅크관리 매커니즘을 응용한 새로운 개념의 캐쉬 구조에 대한 설명을 한다. 크기가 작음에도 불구하고, 낮은 접근 실패율(Miss ratio)와 높은 저전력 효과는 기존의 일반적인 직접사상 캐쉬와 비교했을 때, 성능면에서 월등한 차이를 나타내고 있다. 이러한 결과의 원인은 직접사상 캐쉬와 완전연관 버퍼의 최적화된 구성과, 효과적인 뱅크선별기를 사용하여 적은 전력에도 높은 성능을 발휘하는 새로운 매커니즘을 사용하였기 때문이다. 제안한 구조의 성능은 다양한 크기의 직접사상 캐쉬와 비교하였으며, 접근 실패율, 평균 메모리 접근 시간, 전력소비, Energy * Delay Product 등 모두 4가지의 지표를 사용하였다.

1. 서론

하루가 다르게 기능이 전문화되고 세분화되는 다양한 형태의 정보화기기가 출현함에 따라 기존의 개인용 컴퓨터 등에서만 국한적으로 사용되었던 마이크로 프로세서에 대한 관심이 이제는 고성능 내장형 프로세서에까지 연계 되고 있다. 이러한 내장형 프로세서는 기존의 제어용만을 목적으로 한 것과는 다르게 높은 수준의 계산 능력과 메모리 대역폭, 효과적인 메모리 계층 구조, 메모리 운용 유니트를 통한 가상 메모리 지원 등의 기능을 기본적으로 요구 하고 있는 실정이다. 하지만 무엇보다도 이러한 내장형 프로세서 기능에서 우선적으로 요구되는 것은 전력 소모를 줄이는 데 있다. 저전력은 이러한 내장형 프로세서를 기반으로 제작된 여러 기기들의 사용 시간을 최대한 연장케 하여, 궁극적으로 기기를 사용하는 소비자들의 욕구를 만족시키는데 그 목적이 있다고 할 수 있다.

일반적으로 프로세서를 이루고 있는 다양한 부분 중에서 캐쉬 메모리는 TLB(Translate Lookaside Buffer)와 더불어 전체 칩에서 소모되는 전력의 상당부분을 차지한다.[1] 이는 온칩 메모리 시스템을 구성하는 태그와 데이터 배열들이 프로세서의 빠른 클락 주파수를 지원하기 위하여 주로 전력 소모가 많은 정적 메모리(static RAM)로 구현되기 때문이다. 그리고 이러한 온칩 메모리 시스템은 매우 자주 접근되는 경향이 있어 전력 면에서 볼 때 소비가 매우 많다 할 수 있다. 또, 온칩 메모리 시스템 접근 시에 일어 날 수 있는 실패(miss)는 또 다른 대 용량의 온칩 메모리 시스템을 접근하거나 오프칩 메모리 접근을 위해서 I/O 패드를 구동해야 하므로 전력이 매우 많이 소비된다. 따라서 이러한 캐쉬 메모리들은 전력을 줄이기 위해서 주로 접근 실패율을 줄이기 위한 방향으로 진행 되어오고 있다.

본 논문에서는 내장형 온칩 메모리 참조 시 소비되는 전력을 줄이기 위한 방법으로 뱅크 매커니즘과 효과적인 뱅크의 배분을 위해 뱅크 선별기(bank selector)를 사용하였다. 그 결과 제안된 캐쉬는 DM-16KB에 비해 약 7%의 전력 감소를 효과를 볼 수 있다. 그리고 내장형 온칩 메모리에서 일어나는 접근 실패(miss)를 줄이기 위해 8KB의 직접사상 캐쉬(direct mapped cache)와 1KB 의 완전연관사상 버퍼(fully-associate buffer)를 사용한 결과 접근 실패율(miss_rate)에서는 DM-8KB보다 54%의 성능 향상을 볼 수 있고, 두배 크기인 DM-16KB과는 거의 유사한 성능을 보이고 있다,

이 논문의 나머지 부분은 다음과 같다. 관련 연구는 제 2장에서 소개 되어지며, 제 3장은 제안된 캐쉬의 구조를 설명한다. 제 4장에서는 성능 평가 지표와 소비 전력에 대한 시뮬레이션 결과를 비교-분석한다. 마지막으로 제 5 장에서 결론을 맺는다.

2. 관련 연구

2.1 뱅크 메커니즘 (bank mechanism)

뱅크 메커니즘을 이용하는 캐쉬 또는 TLB는 온칩 메모리 참조 시 소비되는 전력을 줄이기 위한 방법으로 전체 온칩 메모리를 뱅크 구조로 나누는 것이다 [2]. 전체 태그 또는 데이터 부분을 2-뱅크 또는 4-뱅크로 나눌 경우 비트 라인과 워드 라인의 감소에 의한 전력 소비 감소 효과를 얻을 수 있으며, 특히 CAM (content addressable memory)를 이용하는 완전연관 캐쉬의 경우 동시에 참조되는 태그 부분의 엔트리 수가 줄어들기 때문에 소비 전력을 크게 줄일 수 있다. 그러나 이러한 뱅크 메커니즘은 하나의 뱅크에 편중될 확률이 높음으로 다른 뱅크의 활용도 감소로 성능을 저하시키는 단점이 있다

2.2 뱅크 선별기(bank selector)

뱅크 메커니즘의 단점을 줄이기 위한 방법으로, 주소 비트 (Addess bit) 안의 태그 또는 인덱스 비트를 임의적으로 n개씩 비교하여 뱅크를 구별하는 방법이다. 2-뱅크로 나눌 경우에는 태그 또는 인덱스 안의 비트를 임의적으로 4개를 추출해서 AND, OR, XOR-게이트 중 세 개를 선택하여 네 개의 비트를 비교해 해당되는 뱅크를 선택 할 수 있다. 또한, 4-뱅크도 태그 또는 인덱스 안의 비트를 임의적으로 4개 추출해서 AND, OR, XOR-게이트 중 두 개를 사용하여 각각의 비트들을 비교해서 4개의 뱅크 중 해당되는 뱅크를 사용 할 수 있다. 확률적으로 XOR-게이트가 가장 뱅크의 편중을 효율적으로 나눌 수 있으나, 다른 게이트에 비해 전력이 많이 소비된다는 단점을 가지고 있다.

3. 제안된 캐쉬의 구조적 특징과 동작 원리

3.1 제안된 캐쉬의 구조적 특징

새로운 캐쉬의 주된 목적은 크기는 작지만, 빠른 접근 시간(access time), 뱅크를 이용한 저전력 소비 그리고 두 개의 분리된 캐쉬를 선택하여 각각의 특징을 살려 실패율을 최소화하는데 있다. 두 개의 분리된 캐쉬 중 완전연관 버퍼는 2개 혹은 4개의 뱅크로 나누어서 전력 소비를 최소화 하는데 커다란 장점을 가지고, 직접사상 캐쉬는 빠른 접근 시간과 완전연관 버퍼보다 많은 엔트리를 가짐으로써 실패율을 줄이는데 장점을 가지고 있다.

완전연관 버퍼는 크기가 1KB 이고, 2-뱅크로 나눌 때는 각 뱅크당 16개의 엔트리를 갖고, 4-뱅크에서는 각 뱅크당 8개의 엔트리를 갖는다. 메모리참조가 일어나게 되면 뱅크 선별기(bank selector)에서 태그의 두 비트 혹은 네 비트를 비교하여 하나의 뱅크를 선택하게 된다. 선별된 뱅크는 하나의 독립적 완전연관 버퍼처럼 CAM(content addressable memory)을 이용하여 해당되는 메모리는 찾게 된다. 만약 두 개의 캐쉬 모두 접근 실패(miss)가 일어나게 될 때, 완전연관 버퍼 안의 선택된 뱅크 엔트리가 모두 차게 되면, 순차적 방식(FIFO)에 의해서 먼저 저장된 데이터들은 8KB, 전체 256개의 엔트리를 갖는 직접사상 캐쉬로 이동하게 된다. 이때 데이터들은 다시 주소를 재해석하여 메모리 참조 시 완전연관 버퍼와 함께 동시에 참조하게 된다.

3.2 제안된 캐쉬의 동작원리

여기서는 제안된 캐쉬 시스템의 동작원리를 상세히 설명하고자 한다. CPU로부터 메모리 참조가 발생 하게 되면 항상 직접사상 캐쉬와 완전연관 버퍼가 동시에 참조가 일어난다. 가능한 경우로는 다음과 같다

3.2.1 완전연관 버퍼 에서의 적중

메모리 주소가 CPU로부터 발생 되어질 때, 주소의 일부 비트'를 이용하여 캐쉬 내 여러 개의 뱅크들 중에서 하나의 뱅크만 선택하여 사용할 수 있다. 따라서 한번에 하나의 뱅크만 선택 되어짐으로 전력 소비적인 측면에서 유리한 장점을 가지게 된다. 완전연관 버퍼 내의 하나의 뱅크에서 적중이 발생되면 요청한 데이터를 CPU로 보내게 된다.

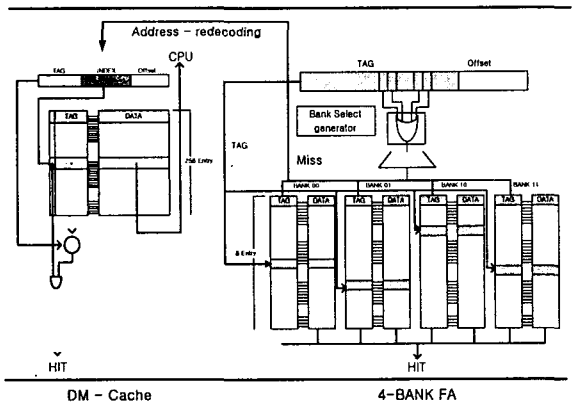

그림 1 제안된 4-뱅크 캐쉬 구조

3.2.2 직접사상 캐쉬에서의 적중

직접사상 캐쉬에서 적중이 발생 되어지면 일반적인 직접사상 캐쉬와 동일한 동작을 수행하게 된다. 즉, 요청 되어진 데이터는 CPU로 보내고 캐쉬의 동작은 끝나게 된다.

3.2.3 두 캐쉬에서 접근 실패

만약 직접사상 캐쉬와 완전 연관 버퍼에서 모두 접근 실패가 발생하면, 완전연관 버퍼의 해당되는 뱅크 엔트리로 접근 실패가 발생한 데이터를 저장한다. 만약 엔트리가 모두 차게 되면 선입선출(FIFO) 알고리즘에 의해 먼저 저장되었던 데이터는 직접사상 캐쉬로 이동하게 된다

4. 성능 평가 지표와 소비 전력

제안된 캐쉬에서 완전연관 버퍼의 최적화된 뱅크분석기를 찾아내기 위해 다양한 시뮬레이션을 수행하였다. 분석적 모델과 시뮬레이션 결과, 4-뱅크에서는 XOR-게이트 두 개가 쓰인 분석기가 가장 좋은 성능을 보였고, 2-뱅크에서는 AND-게이트 세 개가 사용된 뱅크 분석기가 가장 좋은 접근 실패율을 보이고 있다.

4.1 접근 실패율 (Miss_rario)

기존의 일반적인 직접사상 캐쉬와 제안된 캐쉬와의 접근 실패율은 그림 2에서 보여주고 있다. 그림에서 기존의 직접사상 캐쉬는 DM으로 표기하였으며, 기본적으로 모두 32byte의 블록 크기를 가정하고, " K" 는 KB 캐쉬 크기를 나타낸다. 제안된 캐쉬의 "DM8K-FA1K-2bank-aaa"는 8K의 직접사상 캐쉬와 1K의 완전연관 버퍼로 구성된 것 중에서 2-뱅크와 뱅크 선별기로 AND-게이트 3개를 사용한 것을 나타내고 있다. 같은 방법으로 ' x' 는 XOR 게이트를 사용한 것이다. 시뮬레이션은 각각의 벤치마크에 대해 다양한 캐쉬 크기를 변화시켜가면서 수행하였으며 결과적으로 제안된 캐쉬는 일반적인 8K의 직접사상 캐쉬 보다 약 54% 이상의 효과를 보이고, 두 배의 크기인 16K의 직접사상 캐쉬 하고는 유사한 성능을 볼 수 있다. 또한 제안된 캐쉬에서는 2-뱅크가 4-뱅크보다 약간 높은 성능을 볼 수 있다. 벤치마크는 SPEC-2000-int를 사용하였다.

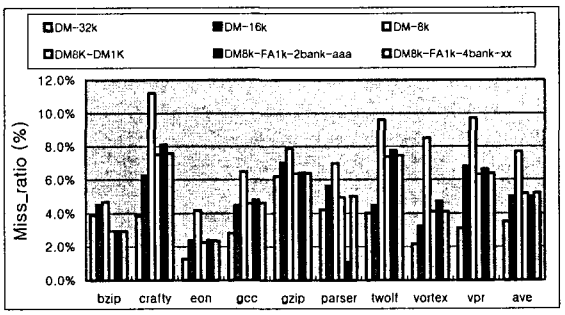

그림 2 접근 실패율

4.2 평균 메모리 접근 시간

캐쉬 시스템의 성능 평가를 보다 정확하게 측정하기 위한 방법으로 평균 메모리 접근 시간을 이용하였다. 평균 메모리 접근 시간을 얻기 위한 식은 다음과 같다.

Average memory access time
= Hit time + Miss rate * Miss penalty. (1)

여기서 hit time은 캐쉬에서 적중을 처리하는데 걸리는 시간이며, miss penalty는 캐쉬 접근 실패 시 이를 처리하는데 추가되는 시간이다. 시뮬레이션을 수행하기 위한 구체적인 변수 값들

은 표 1로써 정의 되어진다. 이러한 변수 값들은 일반적인 32-bit 내장형 프로세서에서 사용 되어지는 값들을 사용하였다.[3]

표 1 시뮬레이션 변수들 (AMAT)

System parameters	Values
CPU clock	200 MHz
memory latency	15 /cpucycle
memory bandwidth	1.6 Gbytes / sec
direct-mapped cache hit time	1 /cpucycle
fully-associative buffer hit time	1 /cpucycle

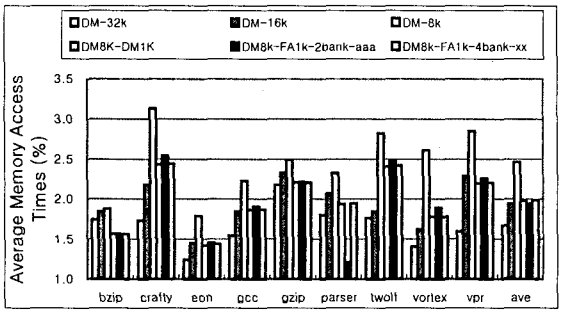

그림 3 평균 메모리 접근 시간

일반적인 직접사상 캐쉬와 제안된 캐쉬와의 평균 메모리 접근 시간에 대한 시뮬레이션 결과는 그림 3과 같다. 제안된 캐쉬의 성능은 그림 2의 접근 실패율과 비교했을 때, 커다란 차이가 없음을 볼 수 있다.

4.3 전력소비 (Power consumption)

메모리 참조가 일어날 때 발생되는 캐쉬의 전체적인 전력을 측정하여 성능을 평가, 비교하였다.[4] 시뮬레이션은 CACTI 3.1로 수행하였으며, 전체적인 기본 측정값들은 표 2에서 정의 되어진다. P_{access}(nJ)과 P_{miss}(nJ)는 적중이 났을 때와 접근 실패가 났을 때의 소비되는 전력을 각각 뜻하고, 제안된 캐쉬의 DM hit은 직접사상 캐쉬에서 적중됐을 때의 소비 전력을 말한다. PAD는 접근 실패가 났을 때, 오프_칩까지 내려가서 다시 데이터를 가져오는 데 소비되는 전력을 뜻하므로[3], 캐쉬에서 접근 실패가 일어났을 때만 소비되는 변수이고, P_{cache_write}는 오프_칩에서 가져온 데이터를 완전연관 캐쉬의 선별된 뱅크 엔트리에 저장하는데 소비되는 전력이다.

표 2 시뮬레이션 변수들 (전력)

Cache configuration	P_{access}(nJ)	P_{miss}(nJ)	P_{cache_write}(nJ)	PAD(nJ)
16KB-32B(DM)	0.4734	0.2230	0.2220	6.48
32KB-32B(DM)	0.6205	0.3735	0.3726	6.48
64KB-32B(DM)	0.9358	0.6918	0.6909	6.48
16KB-32B(2-way)	0.6335	0.2260	0.2237	6.48
16KB-32B(4-way)	0.9349	0.2311	0.2260	6.48
32KB-32B(2-way)	0.7586	0.3426	0.3402	6.48
DM 8KB_FA 1KB-2 bank - aaa	DM hit : 0.4197	0.1660	DM write : 0.1302	6.48
	FA hit : 0.3640		FA write : 0.0779	6.48
DM 8KB_FA 1KB-4 bank - xx	DM hit : 0.4095	0.1558	DM write : 0.1302	6.48
	FA hit : 0.3479		FA write : 0.0779	6.48

표 2에서, 제안된 캐쉬는 전력 면에서 다른 일반적인 캐쉬보다 우수한 성능을 가졌음을 알 수 있다. 특히 제안된 캐쉬의 완전 연관 버퍼에서 적중이 났을 때는 16KB의 직접사상 캐쉬 보다 약 30%, 접근 실패가 일어 날 때에도 약 43%의 전력 감소 효과를 가져오는 것을 볼 수 있다. 그림 4은 표 2의 값들을 기본으로 전체 캐쉬 실패율(miss_ratio)과 적중율(hit_ratio)을 곱한 전체적인 전력소비를 나타낸다.

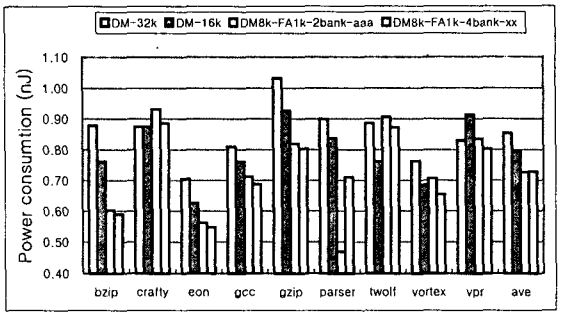

그림 4 전력소비

4.4 Energy * Delay Product

그림 5는 전체 전력소비와 평균 메모리 접근 시간을 곱한 값으로, 수치가 높을수록 전체 성능이 낮아지는 것을 나타낸다. 제안된 캐쉬는 32KB의 직접사상 캐쉬 보다는 높은 성능을, 16K의 직접사상 캐쉬와는 유사한 성능을 가짐을 볼 수 있다.

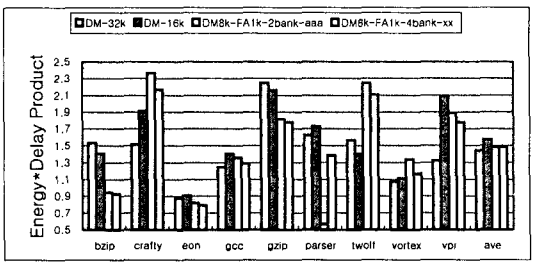

그림 5 Energy * Delay Product

5. 결론

이상 위에서 보는 바와 같이, 제안된 캐쉬는 효율적인 뱅크의 사용과 서로 다른 두 개의 캐쉬 구조의 최적화된 구성으로 인해, 16K의 직접사상 캐쉬와 비슷한 성능을 보이면서도 훨씬 적은 전력을 소비함을 볼 수 있다.

6. 참고문헌

[1] J. Kin, M. Gupta, and W. H. Mangione-Smith, The Filter Cache: An Energy Efficient Memory Structure, MICRO-97: ACM/IEEE International Symposium on Microarchitecture, pp. 184-193, Research Triangle Park, NC, Dec. 1997.

[2] S. Manne, A. Klauser, D. Grunwald, and F. Somenzi, " Low Power TLB Design for High Performance Microprocessors," Univ. of Colorado Technical Report, 1997

[3] J.H.Lee, J.S.Lee, S.W.Jeong, and S.D.Kim, "A Banked-Promotion Performance and Low Power," In ICCD, pp.118-123, 2001

[4] Anita Borg, J.Bradley Chen, Norman P. Jouppi, "A simulation Based Study of TLB Performance," In ISCA, 1991