저전력 캐쉬를 위한 웨이-라인 예측 유닛을 이용한 새로운 드로시 캐싱 기법

論 文
10-2-5

New Drowsy Cashing Method by Using Way-Line Prediction Unit for Low Power Cache

이 정 훈*
Jung-Hoon Lee

Abstract

The goal of this research is to reduce dynamic and static power consumption for a low power cache system. The proposed cache can achieve a low power consumption by using a drowsy and a way prediction mechanism. For reducing the static power, the drowsy technique is used at 4-way set associative cache. And for reducing the dynamic energy, one among four ways is selectively accessed on the basis of information in the Way-Line Prediction Unit (WLPU). This prediction mechanism does not introduce any additional delay though prediction misses are occurred. The WLPU can effectively reduce the performance overhead of the conventional drowsy caching by waking only a drowsy cache line and one way in advance. Our results show that the proposed cache can reduce the power consumption by about 40% compared with the 4-way drowsy cache.

Keywords : Static power, dynamic power, drowsy caching, filter cache, pipeline

I. 서 론

최근에는 고성능의 사양을 갖춘 내장형 시스템이 많이 출시되고 있으며, 특히 아이폰 및 아이패드, 갤럭시탭 등 모바일 기기의 성능은 이미 노트북에 버금가는 고성능의 기능을 갖추었다. 이처럼 시간이 흐를수록 점차 고성능화/고사양의 제품이 출시되고 있으며 프로세서들은 많은 에너지 소비를 요구하게 된다. 그러므로 오랜 시간동안 구동이 가능한 효율적인 시스템에 대한 요구는 꾸준히 증가하고 있지만, 고성능화 시스템의 전력을 공급하는 배터리는 시스템 성능과 비례적으로 성능 개선이 쉽지 않은 실정이다. 온-칩 캐쉬의 경우 프로세서의 소비 전력중 매우 큰 비중을 차지하며 오늘날 프로세서 설계에서 저전력 캐쉬가 가장 중요한 이슈중의 하나로 부각되고 있다[1]. 대표적인 ARM 프로세서인 ARM-920T[2]의 경우 캐쉬 메모리가 차지하는 소비 전력이 44%이며, 가상 메모리를 지원하기 위한 캐쉬의 일종인 TLB (translation lookaside buffer)의 소비전력 또한 9%로 전체 칩의 50% 이상이 이러한 캐쉬 메모리로부터 소비되는 특성을 보이고 있다[2].

또한 고성능화에 따른 미세 공정의 전력 관리에서 누설전력(leakage power)에 대한 비중은 점차 높아지고 있는 추세이며, 누설 전력은 공정기술이 진화함에 따라서 전체 프로세서 소비전력에서 상당부분을 차지하는 주요 요인이 될 것이다. 마찬가지로 시스템 구동에 필요한 동적 (dynamic) 에너지 감소 역시 저전력 설계에서 주요한 부분으로 고려되어야 한다[3].

기존의 대표적인 저전력 캐쉬 기법중 드로시 (drowsy) 기법[4]은 누설 전력을 줄이기 위한 대표적인 기법이라 할 수 있다. 그러나 이 기법의 가장 큰 단점은 드로시 모드(슬림 모드)에 따른 성능 저하를 초래한다. 이에 이러한 단점을 보완하기 위한 새로운 드로시 기법[5]들이 제안되어졌다. 본 연구에서도 이러한 드로시 기법의 단점을

접수일자 : 2011년 04월 15일
심사일자 : 2011년 05월 24일
수락일자 : 2011년 06월 27일
*교신저자, E-mail : leejh@gsnu.ac.kr

극복하기 위하여 새로운 웨이-라인 예측 장치(WLPU: Way-Line Prediction Unit)를 제안하여 캐쉬의 누설 및 동적 소비전력 모두를 낮추고자 한다.

시뮬레이션 결과에 따르면 제안하는 캐쉬 구조는 기존의 대표적인 저전력 캐쉬중 4-way 드로시 캐쉬[4], 4-way 드로시 캐쉥 기법에 필터 캐쉬[6]를 접목한 구조에 비해 약 41% 및 20% 소비전력 감소 효과를 얻을 수 있었다.

II. 관련 연구

최근 에너지 소비는 내장형 시스템이나 프로세서 설계자가 가장 고려해야 할 요인이며 성능의 중요성과 더불어 가장 중요한 설계 요소이다. 그러므로 에너지 소비 감소를 위한 다양한 연구가 선행되었으며 앞으로도 많은 연구가 필요하다. 에너지 소비는 트랜지스터가 스위칭(switching)되면서 발생하는 동적 에너지와 스위칭과는 상관없이 소자 및 공정의 특성상 나타나는 정적 에너지가 있다.

동적 에너지를 줄이는 기법으로 대표적인 구조는 필터구조[6]이다. 필터구조는 L0 케쉬로써 L1 캐쉬위에 작은 캐쉬를 두어 소비전력을 줄이고자 하였다. 그러나 필터구조의 경우 성능 감소의 치명적 단점을 가지고 있다.

블록 버퍼링[7]은 필터 캐쉬와 유사하나 단지 하나의 엔트리만을 가지고 있어 소비전력을 더욱 줄일 수 있는 구조이다. 일단, 한 워드가 캐쉬에 접근을 하게 되면 그 워드를 포함한 하나의 캐쉬 라인이 블록버퍼로 전송이 된다. 만약 다음 실행 때 연속으로 접근을 하게 되면 블록버퍼에서 적중이 일어나고 소비전력을 획기적으로 줄일 수 있으나 접근 실패가 일어나면 정상적인 캐쉬 접근으로 효과가 없다. 이 메커니즘 역시 소비전력적인 측면에서는 필터보다 더욱 효과적인 구조이지만 성능 저하는 더욱 심각한 문제를 야기한다. 그러므로 다중 집합연관 캐쉬처럼 웨이 수에 따른 충분한 버퍼의 제공이 심각한 성능 저하를 막는 방법이라 할 수 있다.

정적 에너지를 줄이기 위한 대표적인 기법들은 주로 공급전압을 줄임으로 해서 누설 전력을 줄이는 방식을 취하고 있다. 이 기법들 중에 [8]은 SRAM 내부의 그라운드단자에 Gated-V_{dd}를 인가해 자주 사용되지 않는 캐쉬 라인의 내부 전압공급을 차단하여 누설 전류를 차단하는 방식이다. 그러나 이 경우 해당 캐쉬 라인의 정보는 소멸되는 단점을 가진다. [9]에서도 일정한 시간동안 쓰이지 않는 캐쉬 라인을 설정해 sleep 모드로 전이시켜 누설전력을 차단하는 기법이다. 이 기법 또한 셀 내부에 저장된 정보를 잃어버리기 때문에 재접근을 할 경우 캐쉬 적중률(cache hit ratio)이 떨어지고 성능저하가 발생하게 된다.

위상 캐쉬(phased cache)[10]는 집합연관 캐쉬에서 태그 비교와 웨이 비교가 동시에 일어나는 과정에서 소비전력을 줄이기 위하여 단계적으로 태그 비교 후 적중이 발생한 웨이만 접근하여 동적 소비전력을 줄이고자 하였다. 그러나 태그 비교를 위한 사이클과 적중 후 하나의 웨이만 접근하는 사이클 동작으로 성능이 크게 저하되는 단점을 보인다.

웨이 예측 캐쉬[11] 역시 집합연관 캐쉬에서 하나의 캐쉬 웨이만 먼저 동작시키고 만약 적중 실패이면 나머지 웨이들을 다음 사이클에 동작시키는 방식을 사용하고 있다. 이러한 예측을 위하여 각각의 셑(set)마다 MRU (Most Recently Used) 알고리즘을 이용하여 해당 셑에 대한 접근시 MRU 비트를 이용하여 최근에 적중이 발생한 웨이를 먼저 접근하는 방식을 사용하고 있다. 이 알고리즘 역시 동적 소비 감소에는 효과적이지만 잘못 예측시 두 사이클이 소비되어 성능 저하를 초래한다.

드로시(drowsy) 기법[4, 5]은 두 가지 상태의 단계를 두고 있다. 먼저 캐쉬에 정상적인 전압을 각각 캐쉬 라인에 공급하는 정상전압 모드(normal mode)와 누설전력을 줄이면서 데이터의 저장 상태를 유지하기 위해 캐쉬 라인에 낮은 전압을 공급하는 드로시 모드(drowsy mode)의 단계를 두고 있다. 캐쉬 라인이 드로시 모드일 때 데이터를 요구할 경우 먼저 캐쉬 라인을 정상상태의 전압으로 승압시켜주는 깨움(wake-up)과정을 거친 후에 접근을 해야 된다. 그러나 이러한 깨움 과정에서 추가적인 사이클이 소비되고 결국 성능 저하의 단점을 내포하고 있다.

본 연구에서는 캐쉬 라인이 드로시 모드일 때 파이프 라인의 한 단계 앞서서 깨워줌으로써 드로시 캐쉬의 가장 큰 단점인 깨움 지연으로 인한

성능저하를 예방 할 수 있다. 이는 예측 테이블이 정상적인 파이프라인의 앞선 단계에 존재하기 때문에 가능하며 또한 예측을 통해 제시된 하나의 웨이만 접근이 가능하여 동적 에너지를 줄여 줄 수 있을 뿐만 아니라 적중 실패 시에도 성능 저하 없이 구동이 가능하다.

III. 웨이-라인 예측 장치

제안하는 웨이 예측 기법을 적용하기 위하여 예측 테이블을 사용한다. 예측 테이블에서는 정확한 예측이 요구되며 테이블의 예측 정확도가 낮으면 모든 웨이를 접근을 해야 하기 때문에 소비 전력측면에서 나쁜 영향을 미치게 된다. 명령어 캐쉬는 웨이 예측을 이용할 때 분기 예측(branch prediction) 기법을 이용할 수 있지만 데이터 캐쉬는 이러한 분기예측 기법을 이용할 수 없고 정확도가 매우 떨어질 수 있다.

제안하는 예측 테이블의 엔트리는 그림 1처럼 태그(Tag), 셀 인덱스(Set index), 웨이 번호(Way number)에 대한 정보를 저장하게 된다.

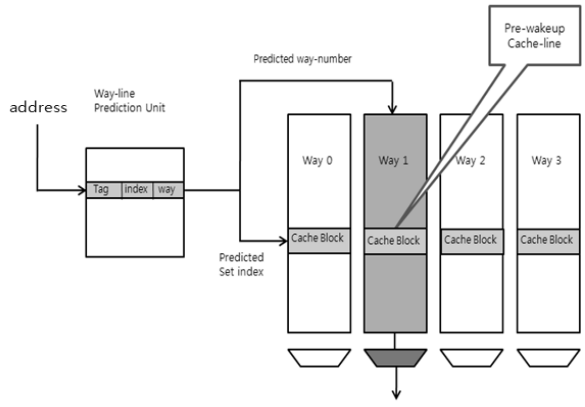

그림 1. 웨이 예측 유닛과 드로시 상태의 캐쉬 라인 깨움
Fig. 1. Way prediction unit and wake-up in drowsy mode

그림 1은 4-way 집합연관 캐쉬 구조와 예측 테이블에 대한 구성도이다. 그림 1처럼 예측 테이블에는 최근에 참조가 일어난 캐쉬 블록에 대한 정보들을 저장하고 있다가 그 라인의 참조가 발생하게 되면 예측 유닛의 태그 값과 생성된 주소의 태그 값 비교 후 적중이면 그 라인만 미리 깨움 동작을 수행하게 되고 정상적인 파이프라인의 해당 단계에서 그 해당 라인만 동작을 시켜 원하는 데이터를 인출 또는 저장이 일어난다. 만약 예측 테이블의 해당 라인에서 태그 적중 실패이면 이는 예측 테이블로 태그 및 웨이의 정보를 알 수 없으므로 드로시 모드인 경우 전체 웨이의 캐쉬 라인을 모두 깨우게 되며, 파이프라인에서 캐쉬 접근시 기존의 집합연관 캐쉬처럼 동일하게 동작하게 된다. 예측테이블은 직접사상 구조(direct mapped structure) 방식으로 구성된다.

그림 2는 웨이-라인 예측 유닛을 이용한 전체적인 파이프라인의 동작 타이밍을 나타내고 있다. 인출/저장을 위한 하나의 데이터가 요청되기 위하여 페치(fetch)가 발생하면 MEM단이 아닌 이슈(issue) 단에서 예측테이블로 접근을 하게 된다. 이때 해당 엔트리 내에 동일 태그 값을 보유하고 있다면 셀 인덱스와 웨이 넘버를 이용하여 해당 캐쉬 라인만을 구동 및 깨울 수 있게 된다. 태그 값이 동일하다는 것은 캐쉬의 태그 값 일치와 동일한 결과로 테이블의 인덱스와 웨이 번호를 통하여 정확한 위치를 알 수 있다. 만약 정상 모드이면 MEM 단계에서 해당 웨이로만 접근을 하여 캐쉬 동작을 수행하게 되고, 드로시 모드의 경우 정확한 위치를 캐시 태그 메모리 전에 알 수 있으므로 그 해당 라인만을 앞선 사이클에서 미리 정상전압을 인가하여 깨워 줄 수 있다.

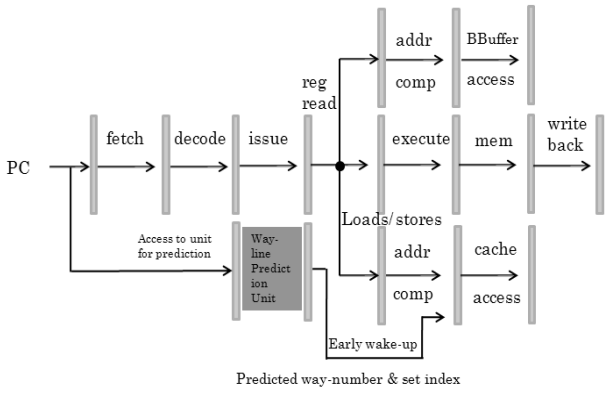

그림 2. 웨이-라인 예측 유닛을 이용한 파이프라인
Fig. 2. Pipeline mechanism by using way prediction unit

그러나 만약 예측 테이블의 해당되는 엔트리 태그가 불일치인 경우 정상모드인 경우에는 기존의 캐쉬처럼 MEM 단계에서 4-way 집합연관 캐쉬의 접근과 동일하게 구동이 일어나고, 드로시 모드의 경우 앞선 사이클에서 모든 웨이를 정상 전압을 인가하여 깨워 놓은 후 MEM 단계에서 기존의 4-way 캐쉬처럼 병렬 탐색을 수행하게 된다. 이때 한 웨이에서 적중이 발생하면 예측 테이

블에 정보를 추가하여 다음 참조에 대한 정보를 보유하게 되고, 적중 실패인 경우에도 L2 캐쉬 또는 메모리로부터 인출이 발생하고 하나의 웨이에 저장이 일어나면 해당하는 태그 값과 인덱스 값, 그리고 저장된 웨이 번호의 정보를 예측테이블에 업데이트 시키게 된다.

IV. 실험 및 성능 평가

1. 실험환경

제안된 구조 및 메커니즘을 평가하기위하여 대표적인 미디어벤치마크[12]를 사용하였다. 알파(alpha) 바이너리 코드로 컴파일을 수행하였으며 Simplescalar 프로세서 시뮬레이터를 사용하여 각각 1억개의 명령어를 수행하는 동안 데이터 참조 주소 및 횟수를 모니터링 하였고, 생성된 트레이스들은 DineroIV 캐쉬 시뮬레이터를 수정하여 성능을 평가하였다. 실험을 위한 시스템 구성 파라미터는 표1에서 보는 것과 같다. 소비전력을 측정하기 위해서 Cacti[13]시뮬레이터를 사용하였으며 동적에너지 소비는 표 2와 같다.

표 1. 시스템 파라미터
Table 1. System parameter

parameter	value
CPU	2 issue per cycle
L1 I-cache	32KB, 4way, 32byte block, 1cycle latency
L1 D-cache	32KB, 4way, 32byte block, 1cycle latency
L2 cache	none
memory	30 cycle latency
Prediction Unit for way and wake-up	16,32,64,128,256, 512,1024
windows	2048

2. 성능 평가

다양한 저전력 기법 및 구조를 이용하여 정적에너지와 동적에너지 소비를 고려하여 실험을 하였다. 표 2는 캐쉬에 접근할 때 필요한 동적 에너지를 나타낸다. 기본 구조는 4-way 집합연관 캐쉬이며, 웨이 예측 캐쉬 구조, 블록버퍼 구조, CAM 기반의 필터 캐쉬 구조, 그리고 제안된 구조에 대하여 수행하였다. 제안된 웨이-라인 예측 장치(WLPU)의 엔트리는 미디어벤치마크의 특징

표 2. 각 구조에 대한 동적 에너지
Table 2. Dynamic power for each configures

scheme	energy
block buffer	1.44pJ
4-way set-assoc	32.4pJ
4 filter	5.87pJ
16 entry Prediction	0.40pJ
32 entry Prediction	0.42pJ
64 entry Prediction	0.47pJ
128 entry Prediction	0.60pJ
256 entry Prediction	0.71pJ
512 entry Prediction	0.91pJ
1024 entry Prediction	1.26pJ

에 따라 달라진다. 본 실험에서는 WLPU의 엔트리 수를 16에서 최대 512의 엔트리까지 늘려가면서 수행하였다.

제안된 캐쉬 구조에서 그림 3에서 보듯이 동적 에너지가 WLPU의 크기가 증가함에 따라 감소하고, 정적 에너지는 WLPU의 크기가 증가함에 따라서 조금씩 증가하고 있다. 이는 WLPU가 증가 하면서 예측을 통한 한 웨이의 접근 빈도 증가로 동적 에너지를 감소시키지만, 반대로 예측 테이블의 엔트리가 증가를 하면서 누설전력은 오히려 증가하고 있는 것이다.

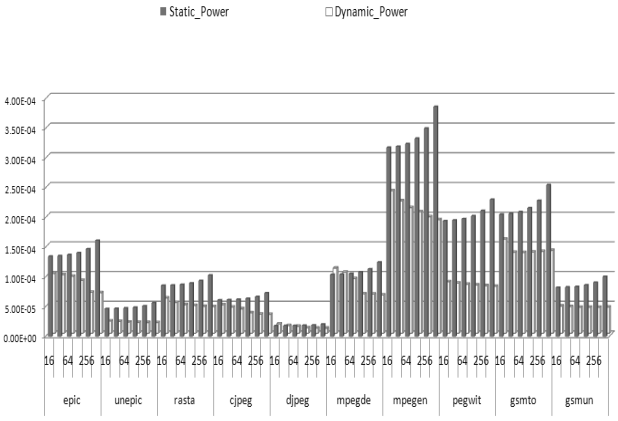

그림 3.엔트리에 따른 WLPU 캐쉬의 정적/동적 소비 전력 (J)
Fig. 3. Static/dynamic power consumption of WLPU cache with various entries

그림 4는 본 논문에서 제안한 구조와 CAM기반의 필터 캐쉬(filter cache), 웨이 예측(way prediction) 캐쉬 그리고 블록 버퍼를 적용한 4-way 캐쉬 구조에 대한 소비전력이다. 기준 캐쉬는 4-way 캐쉬 구조에 드로시 기법을 적용한 구조로써 이를 기준으로 정량화 하였으며 비교 캐쉬들의 모든 구조들도 기존의 드로시 기법을 적용한 결과이다.

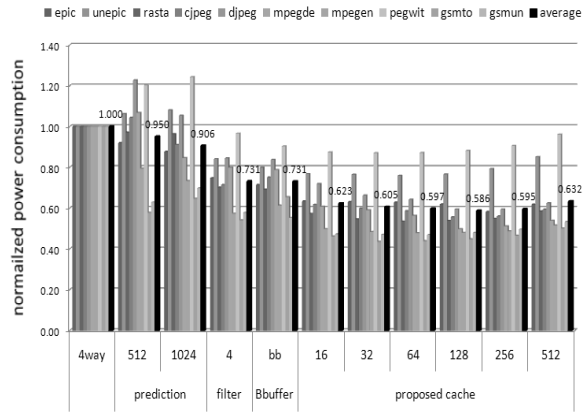

그림 4. 제안된 캐쉬 메커니즘과 다양한 비교 구조와의 정량화한 소비 전력 비교

Fig. 4. Power comparison of proposed WLPU and other caches

제안된 구조의 경우 평균화한 수치는 WLPU의 엔트리가 128개에서 가장 작게 나타난다. 이것은 WLPU의 엔트리 개수가 증가하면서 비례적으로 증가하지 않음을 알 수 있고, 128개의 엔트리를 초과하여 구성할 경우 비용적인 측면뿐만 아니라 오히려 소비전력 증가를 가져오게 된다.

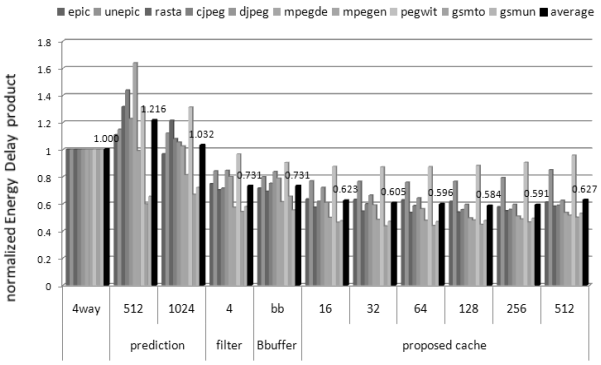

그림 5. 제안된 캐쉬 메커니즘과 다양한 비교 구조와의 정량화한 에너지-딜레이 곱

Fig. 5. Energy-delay product comparison of the proposed WLPU cache and other caches

그림 5는 본 논문에서 성능과 에너지의 성능을 하나로 나타내는 지표인 에너지*성능 곱 ED (Energy-Delay Poduct)에 대한 시뮬레이션 결과이다. 그림 4와 5에 대하여 시뮬레이션 결과를 살펴보면 필터 캐쉬 및 블록 버퍼는 CAM기반의 4개 완전연관 캐쉬(fully set associativity cache)로 구성을 하여 에너지 소비를 측정을 하였다. 두 캐쉬 구성은 벤치마크별로 조금씩 다른 성능을 보이고 있지만 평균적으로는 거의 비슷한 결과값을 보여주고 있다. 이는 필터 캐쉬는 블록 버퍼에 비해 접근 에너지가 다소 높고, 누설전력이 많이 발생하지만, 블록 버퍼보다 필터 캐쉬로의 접근이 더 많이 이루어진다. 두 경우 모두 약 27%의 에너지 감소 효과가 나타났다. 제안된 캐쉬에서는 WLPU의 엔트리가 128개 일 때 가장 많은 에너지 소비 감소를 보여주는데, 평균적으로 약 41% 정도의 에너지 감소 효과를 얻을 수 있었다. 하지만 제안된 구조에서 WLPU의 엔트리가 16, 32, 64의 개수만 사용하더라도 충분히 좋은 에너지 감소효과를 얻어 낼 수 있기 때문에, 많은 엔트리를 사용하여 예측 장치를 구성하지 않고 적은 수의 엔트리를 이용해서도 충분히 소비 전력을 감소시킬 수 있다.

그림 5의 ED 역시 128개의 엔트리에서 가장 좋은 결과를 얻어 수 있었다. ED 역시 소비 에너지와 비슷한 결과가 나오는데 이것은 제안된 캐쉬의 구성이 성능적인 감소가 나타나지 않는다는 것이다. 웨이 예측 캐쉬의 경우 성능의 저하로 기존의 드로시 4-way 구조보다 오히려 ED가 더 떨어지고 있으며, 나머지 구조들은 그림 4와 큰 차이가 없음을 알 수 있었다.

V. 결 론

오늘날 내장형 프로세서내의 연관사상 캐쉬로 인해 나타나는 동적 에너지 소비 증가와 공정기술의 발달에 따른 정적에너지 소비가 증가 하고 있다. 본 논문은 드로시(drowsy) 기법을 이용하고 WLPU를 이용하여 데이터 캐쉬의 동적에너지와 정적에너지를 줄인다. 모든 웨이-라인을 병렬로 접근하는 것을 하나의 웨이만 접근시킴에 따라 동적 에너지를 크게 감소시킬 수 있다. 또한 드로시 기법으로 자주 쓰이지 않는 캐쉬 라인을 저전압상태로 전이시켜 정적 에너지를 줄이고, WLPU에서 드로시 캐쉬라인에 접근할 경우 한 단계 앞서 파이프라인에서 깨움 동작이 일어남으로 성능 저하를 없앨 수 있었다. 시뮬레이션 결과에 따르면 WLPU 128엔트리에서 가장 좋은 결과를 보이고 있는데 4-way 드로시 캐쉬에 비해 약 41%의 전력 소비가 감소되었으며 작은 용량의 32개 엔트리만으로도 매우 우수한 결과를 얻을 수 있다.

[참 고 문 헌]

1] P. Petrov and A. Orailoglu, "Dynamic Tag Reduction for Low-Power Caches in Embedded Systems with Virtual Memory," *International Journal of Parallel Programming*, vol. 35, no. 2, pp. 57-177, 2007.

[2] S. Segars, "Low power design techniques for micro-processor," *Proceedings International Solid-State Circuits Conference Tutorial,* 2001.

[3] R. Komiya, K. Inoue, and K. Murakami, "Dynamic Management Technique to Mitigate Performance Degradation for Low-Leakage Caches," *Proceedings IEEE Symposium on Low-Power and High-Speed Chips: Cool Chips X*, 2007.

[4] S. Segars,"Low Power Design Techniques for Microprocessors," *Proceedings IEEE International Solid-State Circuits Conference*, 2001.

[5] J. Zushi, , G. Zeng, H. Tomiyama, H. Takada, and K. Inoue, "Improved Policies for Drowsy Caches in Embedded Processors," *Proceedings IEEE International Symposium on Electronic Design, Test and Applications*, 2008.

[6] J. Kin, M. Gupta, and W. H. Mangione-Smith, "The Filter Cache: An Energy Efficient Memory Structure," *Proceedings International Symposium on Microarchitecture*, pp. 184-193, 1997.

[7] J. Lee, C. Weems, and D. Kim,"Selective block buffering TLB system for embedded processors," *IEE Proceedings Computers and Digital Techniques*, vol. 152, no. 4 , pp. 507-516, 2005.

[8] M. Powell, S. Yang, B. Falsafi, K. Roy, and T. Vijaykumar, "Gated-vdd: A circuit technique to reduce leakage in deep-submicron cache memories," *Proceedings of the International Symposium on Low Power Electronics and Design*, 2000.

[9] S. Kaxiras, Z. Hu, and M. Martonosi, "Cache decay: Exploiting generational behavior to reduce cache leakage power," *Proceedings of the 28th International Symposium on Computer Architecture*, 2001.

[10] A. Hasegawa, I. Kawasaki, K. Yamada, S. Yoshioka, S. Kawasaki, and P. Biswas, "SH3: high code density, low power," *IEEE Micro*, pp. 11-19, 1995.

[11] I. Koji I. Tohru Ishihara and M. Kazuaki, "Way-Predicting Set-Associative Cache for High Performance and Low Energy Consumption," *Pro. ISLPED'99*, 1999.

[12] C. Lee, M. Potkonjak and W. Mangione-Smith, "Mediabench: A tool for evaluating and synthesizing multimedia and communications systems," *Proceedings of the 30th International Symposium on Microarchitecture*, 1997.

[13] Cacti simulator: http://www.hpl.hp.com/research/cacti/

Biography

이 정 훈

1999년 성균관대학교 제어계측공학과 졸업
2001년 연세대학교 컴퓨터과학과(공학석사)
2004년 연세대학교 컴퓨터과학과(공학박사)
2004년~현재 국립경상대학교 제어계측공학과 부교수

<관심분야> Embedded system, Microprocessor, Low power, SOC system
<e-mail> leejh@gsnu.ac.kr