

메모리 특성을 반영한 동적전압변경 기법

최진욱^o 차호정
연세대학교 컴퓨터과학과
{jwchoi, hjcha}@cs.yonsei.ac.kr

A Memory-Aware Dynamic Voltage Scaling

Jinuk Choi^o Hojung Cha
Dept. of Computer Science, Yonsei University

요 약

배터리의 용량이 제한적인 핸드헬드 시스템에서의 주요 전력소비원인 프로세서의 전력소비를 절감시키기 위한 동적전압 변경 기법에 관련한 많은 연구가 이루어지고 있으나, 프로세서에 집중된 DVS는 MPEG클립 재생과 같은 멀티미디어 응용 프로그램에 있어서 빈번히 발생하는 프로세서와 주 메모리 간의 상호작용은 충분히 설명하지 못하고 있다. 본 연구에서는 주 메모리와 프로세서간의 상호작용을 반영한 동적전압변경을 제시하고 실제 시스템에서의 구현 결과를 제시함으로써 메모리인지 동적전압변경 기법이 멀티미디어 응용에 있어서 고전적 DVS방법보다 QoS를 유지하면서 부가적인 전력 절감을 할 수 있음을 보인다.

1. 서론

동적전압변경 기법은 PDA와 Notebook과 같은 휴대용 디바이스에서 배터리의 수명을 연장시키기 위한 기법중의 하나로, 시스템의 전력 소비의 많은 부분을 차지하는 프로세서의 동작주파수를 수행되는 응용프로그램의 수행 복잡도에 따라 능동적으로 변화시키는 기법이다. 동적전압변경기법은 과거의 작업량을 가지고 다음 프로세서의 성능을 측정하는 고전적 방법에 대표적인 Weiser[1] 와 Govil[2] 으로부터 발전하고 있다. 이러한 프로세서 작업량 기반 예측(processor utilization based prediction)은 프로세서의 작업들의 많은 부분이 프로세서의 코어에서 수행되는 태스크 모델에는 적합하나, 외부 주 메모리의 사용량이 많은 멀티미디어 응용프로그램에서는 적합하지 않다.

이러한 문제를 Dirk[3]는 실험적으로 언급하였으나 비선형적 변칙(non-linearity abnormalities)이란 용어를 사용하면서 단순한 관찰에 머물렀고, Martin[4]은 더욱 다양한 실험으로 프로세서와 메모리에 상관관계가 빈번하다는 점에 대한 관찰을 제시하였다. 하지만 이러한 현상의 원인을 정확하게 파악하지 못함으로써 어떻게 DVS에 적용하는가에 대해서는 접근을 하지 못하고 있다. Xiaobo[5]는 프로세서와 메모리의 상호작용관점에서의 저전력 운영방안을 제시 하였으나, 검증에 있어서 시뮬레이션을 사용함으로써 실제 시스템에서 프로세서, 캐쉬 메모리, 주 메모리의 상호 작용에서 발생하는 비선형적인 환경을 반영하지 못하고 있다.

본 논문에서는 핸드헬드 시스템에서 멀티미디어 응용 프로그램 수행 시 프로세서와 메모리의 상관관계에 의한 메모리 월(memory wall) 문제가 존재함을 보여주고, 이 문제를 해결할 수 있는 방안을 제시한다. 또한 실험을 통해 제안된 시스템의 성능을 분석한다.

2. Memory Wall 문제

메모리 월 문제는 메모리의 성능 발전 속도가 상대적으로 프로세서의 성능증가 보다 느려서, 메모리 서브시스템이 전체 시스템의 성능에 병목현상을 초래한다는 컴퓨터 구조론에 있어서의 고전적 문제이다. 이 문제는 핸드헬드 시스템에서는 더욱 더 심화되는데, 이는 핸드헬드 시스템에서는 데스크톱 컴퓨터와 노트북에서 최근 사용되고 있는 램버스 디램과 DDR과 같은 고성능 메모리를 전력상의 문제로 채택하지 못하기 때문이다. 메모리 월 문제는 주 메모리를 많이 사용하는 멀티미디어 응용에서 더욱 심화된다. 이 현상을 실험적으로 보이기 위하여, 300kbps, 500kbps, 700kbps, 와 900kbps의 MPEG 클립들을 임베디드 리눅스가 탑재된 인텔 PXA270[6] 개발 시스템에서, 프로세서의 동작 주파수를 117Mhz 로부터 416Mhz 까지 16단계로 선형적으로 변화시켜가며, mplayer로 재생하였다. 그림1은 커널상의 스케줄러에서 시스템 타이머를 모니터링 하고 이 값을 기반으로 계산한 시스템 사용량을 보여준다.

그림에서 관찰할 수 있는 것은 프로세서의 동작 주파수가 선형적으로 증가하더라도 프로세서의 사용량(프로세서 성능의 역수)은 선형적으로 감소하지 않는다는 것이다. 특히, 프로세서의

• 본 연구는 정보통신연구진흥원에서 지원하는 정보통신기초기술연구지원사업으로 수행하였음 (과제번호 : 04-기초-065)

동작 주파수가 234Mhz의 경우에는 프로세서 사용량이 평균에 비하여 4배 가까이 증가되는 현상을 보여준다. 이 현상은 특정부분에서의 메모리의 성능이 상대적으로 프로세서의 성능을 따라가지 못하는 메모리 윌 문제가 멀티미디어 응용에서 두드러짐을 보여준다. 즉, 고전적 DVS 기법이 가정하는 프로세서 성능이 주파수에 비례한 선형적 증가와 이에 따른 새로운 i 단계에서의 프로세서의 퍼포먼스(P_i) 설정값이 전 단계에서의 프로세서 사용량(U_{i-1})에 비례한다는 ($P_i = kU_{i-1}$, k 는 스케일링 상수) 가정이 모순이 된다는 것을 보여준다. 본 논문에서는 메모리 윌 문제를 감지할 수 있는 새로운 메트릭을 제시하고 메모리 윌 문제를 보상할 수 있는 메모리 인지(memory aware) DVS를 제안한다.

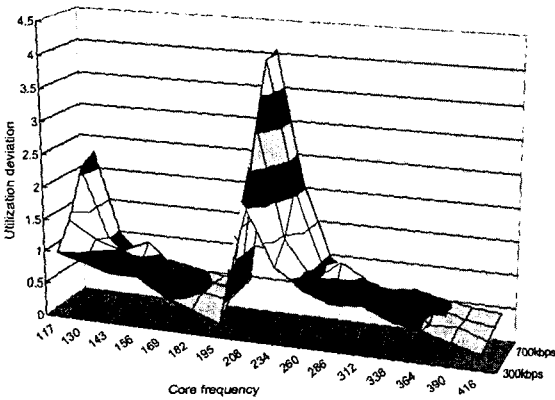


그림1. MPEG 응용프로그램수행시의 메모리 윌 문제

3. Memory-Aware DVS 시스템

최근에 개발되는 핸드헬드 시스템에서 사용되는 프로세서는 SOC에 기반을 두어 데스크톱 컴퓨터에서 수행되던 응용프로그램들을 소형 디바이스에서도 수행케 하고 있다. 동시에 성능을 보장하고 분석하기 위한 전처리 작업인 성능모니터링하기 위한 기능을 포함시켜 소프트웨어 적으로 내부의 행동 방식을 알 수 있는 장치들이 보편화 되고 있다. 구체적으로 이러한 장치를 PMU(Performance Monitoring Unit) 이라 부르며 예전에 시뮬레이션에서나 가능했던, 데이터 캐쉬, 인스트럭 캐쉬, TLB등에 관련한 정보를 실제 프로세서의 동작중에 읽어 낼 수 있다.

본 연구에서는 메모리 윌 현상이 주 메모리 접근에 관련된 문제임에 착안하여, 외부 주 메모리의 액세스를 빈번하게 일으키는 데이터 캐쉬의 미스 레이트를 모니터링 하여 메모리 윌 문제를 일으킴을 판단할 수 있음을 가정하고 실제시스템에서 이를 관찰하였다. 그림2는 메모리 윌 현상이 일어났던 부분에서 동일하게 데이터 캐쉬의 미스 레이트가 과도적으로(이번 실험에서는 평균보다 5배정도) 발생하는 것을 보여준다. 이러한 관찰에 기반을 두어, 본 논문에서는 이 메트릭을 고전적 DVS에 부가적으로 사

용하여 메모리 윌 문제를 보상하는 메모리 인지 동적전압변경기법(Memory-Aware DVS; 이하 M-DVS)을 제시한다.

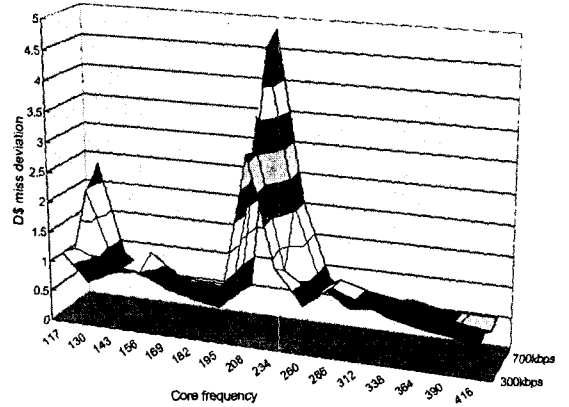


그림2. MPEG 응용 프로그램수행시의 데이터 캐쉬 미스 레이트

다음은 M-DVS를 수행하기 위한 2단계 알고리즘을 제시한다. 1단계에서는 고전적 DVS에서 많이 사용하는 PAST[1] 기반의 DVS를 이용하여 휴리스틱기법에 의한 전력 최적 값을 찾고, 단계 2에서는 앞서 보인 데이터 캐쉬의 미스 레이트의 메트릭을 가지고 메모리 윌 문제를 보상하는 방법을 사용한다. 이 기능을 수행하는 의사 코드는 다음과 같다.

```

M-DVS(phase){
  if(phase==1){
    for(K=Windowmax/2 to 100){
      set DVSwindow=[ 0 K/4 K/2 3K/4 K]
      select K* which minimizes power of total clips
    }
    return;
  }else if(phase==2){
    Uthr = K*
    for(D$Thr=D$miss lower bound to D$miss upper bound)
      select D$Thr* which minimizes power of total clips
  }
  DONE 2-phase:
  DO_DVS(K*,D$Thr*)
}
    
```

1 단계에서는 고전적 DVS에 있어 퍼포먼스 책정을 결정하는 임계값들의 집합인 DVS_{window} 를 변경시켜 가면서 전력소비를 최소화 하는 K^* 값을 선정한다. 이 휴리스틱 최적 값을 바탕으로 단계 2에서는 $D\$_{Thr}$ 를 변경시켜 가면서 전력 측정값을 최소화

하는 $D\$_{Thr}^*$ 를 정한다. 임계값 설정 과정이 끝나게 되면 이 값들을 기반으로 DVFS를 수행 하게 된다.

```
DO_DVS( $K^*$ ,  $D\$_{Thr}^*$ ){
    if( $U_{i-1} < K^*/4$ )  $P_i = f_{max}/4$ 
    //  $f_{max}$  refers to maximum operating frequency of the
    // processor
    else if( $U_{i-1} < K^*/2$ )  $P_i = f_{max}/2$ 
    else if( $U_{i-1} < K^*3/4$ )  $P_i = f_{max}3/4$ 
    else if( $U_{i-1} < K^*$ )  $P_i = f_{max}$ 
    if( $D\$_{miss\ rate} > D\$_{Thr}^*$ )  $P_i = P_i - 1$ 
}
```

4. 실험

다음은 실제 내장형 시스템에서 제시한 M-DVS 기법이 기존의 고전적방법의 DVS보다 더 전력을 절감할 수 있음을 보인다. 실제 임베디드 시스템은 인텔 PAX270 프로세서가 내장된 개발기기를 사용하였고, 운영체제는 임베디드 리눅스 2.4.19를 탑재하였다. 프로세서의 사용량과 내부 캐쉬 매트릭을 측정하고 계산하기 위한 모니터링 유닛과 M-DVS 알고리즘은 동작상의 오버헤드를 줄이기 위하여 커널상의 스케줄러에 구현하였다. 프로세서의 동작 주파수와 전압을 바꾸기 위한 함수들도 API들로 구성 하였다. 실제 시스템의 전력 측정은 Agilent사의 자료 수집기(data logger)인 34970A/34901A를 사용하여 시스템 프로세서에 실제 공급되는 전압과 전류를 동시에 측정하여 전력계산의 정도를 높였다($P_{sampled} = V_{core} * V_{drop} / R_{drop}$; $P_{sampled}$: 샘플 시간 내의 전력 소비량, V_{core} : 프로세서 코어에 공급전압, V_{drop} : 저항에서의 전압 강하, R_{drop} : 전류 측정용 저항). 표 1 은 300kbps, 500kbps, 700kbps 와 900kbps의 MPEG 클립들을 mplayer로 재생 하였을 때의 DV S를 하지 않았을 경우와 PAST-DVS, M-DVS 정책들의 전력소비 값들과 상대 전력절감 정도를 나타낸다.

표1. MPEG 클립 재생 시 정책에 따른 프로세서 전력소비량

Policies MPEG Workloads	Non-DVS	PAST-DVS	M-DVS
	300kbps	0.241	0.075(69%)
500kbps	0.267	0.103(61%)	0.078(71%)
700kbps	0.250	0.099(61%)	0.082(67%)
900kbps	0.261	0.131(50%)	0.098(62%)
Units	Average power consumption in W (Relative reduction to Non-DVS in %)		

표 1 에서 보는 바와 같이 PAST-DVS는 각각의 bit rate에 대하여 DVS를 사용하지 않았을 경우보다 각각 69%, 61%, 61%, 50%, 평균적으로 60%정도 전력을 절감할 수 있음을 보인다. 반

면에 M-DVS는 각각 76%, 71%, 67%, 62%, 평균적으로 약69%정도의 전력 절감효과를 보인다. 900kbps의 경우에는 다른 비트 레이트의 클립보다 상대적으로 PAST-DVS에 비하여 M-DVS의 전력절약 효과가 두드러지는데, 이는 M-DVS 알고리즘이 주 메모리의 활용도가 높은 900kbps 클립에서 더욱 잘 작용한다고 해석된다. 본 실험결과는 MPEG의 중요한 QoS 요소인 drop frame을 모니터링한 결과 제시된 실험 조건 내에서는 PAST-DVS 와 M-DVS 정책적응에 의한 QoS의 감소가 없음을 확인 하였다.

5. 결론

핸드헬드 디바이스의 소형 경량화와 성능 향상의 두 가지 목적은 제한된 배터리의 용량을 효율적으로 사용하기 위하여 DVS라는 기법의 연구를 활성화 시켰다. 하지만 프로세서만 고려한 고전적 DVS는, 현재 소형 디바이스에서 중요한 응용중인 하나인 멀티미디어 응용에 있어서는 시스템 은 칩의 구조적 한계 때문에 발생하는 컴퓨터 구조의 고전적인 메모리 월 문제를 심각하게 직면하게 된다. 본 연구는 이러한 문제를 단순히 관찰하고 비선형적인 문제로 간과한 기존연구에서 나아가 메모리 월 문제를 실제 시스템에서 실험적으로 재현하고, 이것을 극복하기 위한 매트릭을 찾아내어 메모리를 고려한 M-DVS를 제안 하였다. 또한 실제 임베디드 시스템에서 구현함으로써 전력절감에 한계에 다다른 고전적 DVS에서 아무 QoS의 손상 없이 부가적 전력 절감을 이루어 냈다는 것을 보였다.

참고문헌

- [1] M Weiser, B. Welch, A. Demers, S. Shenker, "Scheduling for reduced CPU energy," *Proc. of USENIX Symposium on Operating Systems Design*, pp.13-23, Nov. 1994.
- [2] K. Govil, E. Chan, and H. Wasserman, "Comparing algorithms for dynamic speed-setting of a low-power CPU," *Proc. of ACM Int'l Conf. on Mobile Computing and Networking*, pp. 13-25, Nov. 1995.
- [3] Dirk Grunwald, et al., "Policies for Dynamic Clock Scheduling," *Proc. of the 4th USENIX Symposium on OSDI00*, Oct. 2000.
- [4] T. L. Martin, *Balancing Batteries, Power and Performance: System Issues in CPU Speed-Setting for Mobile Computing*, Ph.D. thesis, Dept. of Electrical and Computer Engineering, Carnegie Mellon University, 1999.
- [5] Xiaobo Fan, Carla S. Ellis, Alvin R. Lebeck, Duke University, "The Synergy between Power-aware Memory Systems and Processor Voltage Scaling," *Proc. of Power Aware Computer Systems (PACS'03)*, Dec. 2003.
- [6] Intel, *intel PXA27x processor family*, developer.intel.com/design/pc a/prodbref/253820.htm, 2004.