Windows CE 환경에서 PXA320 프로세서의 DVFS를 위한 성능 모니터링

심재원°, 이상정
순천향대학교 컴퓨터공학부
e-mail: piring@gwbs.net, sjlee@sch.ac.kr

Performance Monitoring for DVFS of a PXA320 Processor in the Windows CE Environment

Jae-Won Shim, Sang-Jeong Lee,
Dept of Computer Science and Engineering, Soonchunhyang University

요 약

본 논문은 성능 카운터를 이용하여 Intel XScale 마이크로아키텍쳐 기반의 Marvell PXA320 프로세서에 대한 성능 모니터링을 구현하였다. Windows CE 운영체제 환경의 응용프로그램에 대하여 DVFS 구성에 따른 각각의 벤치마크를 측정하였고, 성능 이벤트에 따른 성능 카운터 값을 측정 하였다. 성능 모니터링으로 측정된 데이터를 기반으로 DVFS 기법을 위한 스케줄링이 가능하다.

1. 서론

최근의 컴퓨터 시스템의 설계와 관리에서 전력 및 에너지는 컴퓨터 성능과 같은 수준의 주요 관심사이다. 에너지의 효율성은 모바일과 임베디드 시스템의 중요한 특성이다.

모바일과 임베디드 시스템과 같은 배터리 기반의 장치들은 데스크탑 장치에 비하여 더 효율적인 전력관리를 요구한다. 최근의 강력한 모바일 프로세서의 에너지 요구증가와 짧은 배터리 수명, 발열로 인한 에너지 손실 등으로 인한 전력 소모 때문에, 전력과 에너지 소모를 줄이기 위한 많은 연구가 진행되고 있다. 배터리 기반 장치들의 전력 소비량을 고려하는 하드웨어와 소프트웨어의 설계가 필요하다.

전력과 에너지 소모를 줄이기 위한 전력 관리 기법 중 하나인 DVFS(Dynamic Voltage/Frequency Scaling)는 시스템의 공급전압과 클럭 주파수를 변경하여 마이크로프로세서 및 그 외 회로의 전력소모를 조정하는 방법을 제공한다. DVFS는 공급 전압과 에너지 소비 사이의 이차방정식을 가지며, 여러 가지 DVFS 구성에 따른 시스템의 사용률과 전력소모를 분석하여, 전력소모를 줄이기 위해 필요한 DVFS 기법에 대한 스케줄러를 제작할 수 있다. 전압의 일정한 감소는 전원을 배로 절약할 수 있게 하며, 이를 통해 제한된 용량의 배터리가 달린 모바일 장치로 하여금 유효전원을 좀 더 확보할 수 있게 한다.*

* “이 논문 또는 저서는 2007년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임” (KRF-2007-지역대학우수과학자지원사업-D00394)

성능 모니터링 카운터는 프로세서와 메모리 시스템 활동에 대한 런-타임 가시성을 제공하여 관리목표를 충족하는 DVFS 상태를 선택하는 지능적인 결정을 돕는다. 또한 응용프로그램 실행 중에 전력동작을 알 수 있으며 소프트웨어의 성능 특성을 변경하는데 사용될 수 있다. 성능 카운터는 메모리 액세스, 파이프라인 스톨 같은 수많은 이벤트 감시를 지원한다. 성능 카운터로 갱신된 데이터를 분석하여 DVFS에 필요한 전력소모 예측이 가능하며, 이를 위하여 프로세서의 클럭 주파수 변경에 영향을 받지 않는 성능 카운터 선택이 중요하다.

본 논문에서는 Marvel PXA320 프로세서를 이용하여 다양한 전압/주파수 구성에서의 응용프로그램에 대한 성능 모니터링을 구현하였다.

MS Windows CE 5.0 운영체제 상에서 DVFS와 성능 모니터링 제어를 위한 드라이버를 제작하였고, 성능 이벤트에 대한 성능 모니터링 데이터를 수집하였다.

2. XScale 마이크로아키텍쳐 기반의 Marvell PXA320 프로세서

본 논문에서는 Marvell PXA320 (Monahans) / ARM 11 프로세서, 128MB DDR RAM, 128MB NAND Flash를 장착한 Microvision 사의 MV320-LCD 개발보드를 사용하였고, MS Windows CE 5.0 운영체제를 탑재하였다.

Marvell PXA320 프로세서는 Intel XScale 마이크로아키텍체를 기반으로 하고 있으며, 동적 전압/주파수 스케일링을 지원한다. 응용프로그램에서 전압과 주파수의 동적인 변경을 이용하여, 전력소모를 줄이는 성능과 전압에 최적

인 상태로 구성이 가능하다. PXA320 프로세서에서 지원되는 클럭 스피드는 다음과 같다. : 806 MHz, 624 MHz, 416 MHz, 403 MHz, 312 MHz, 208 MHz, 104 MHz

PXA320 프로세서의 코어 클럭은 13MHz이고, 코어 PLL은 2가지의 터보 클럭과 런 클럭을 제공한다. 터보 클럭은 13MHz * XL * XN으로 값이 구해지고, 런 클럭은 13MHz * XL로 구해진다. 여기에서 XL은 코어 PLL의 터보모드와 런모드의 비율(1:1, 2:1)이고, XN은 코어 PLL의 런모드와 진동자의 비율(8:1, 16:1, 24:1, 31:1)이다. 표.1 에서는 클럭 주파수와 쌍을 이루는 공급전압과 그에 따르는 런모드/ 터보모드를 보여준다.

표. 1 PXA320의 주파수와 전압 설정

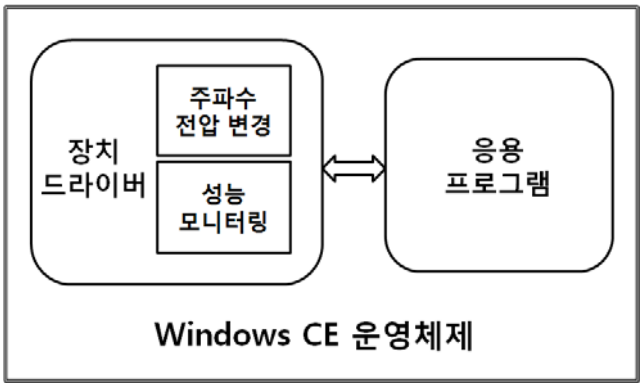
주파수	공급 전압	Run / Turbo Mode
104 MHz	1 V	Run Mode
208 MHz	1 V	Run Mode
312 MHz	1.375 V	Run Mode
403 MHz	1.4 V	Run Mode
416 MHz	1.1 V	Turbo Mode
624 MHz	1.375 V	Turbo Mode
806 MHz	1.4 V	Turbo Mode

Intel XScale 마이크로아키텍처에서는 2가지 형태의 성능 모니터링을 지원한다. 첫번째 성능 모니터링 유닛은 2개의 32비트 이벤트 카운터와 하나의 32비트 주기 카운터로 이루어지며, 두번째 성능 모니터링은 4가지 고유의 이벤트를 동시에 모니터 할 수 있는 4개의 32비트 성능 카운터와 하나의 코어 클러 주기를 카운트하는 32비트 주기 카운터로 이루어진다.

PXA320 프로세서에서는 4개의 성능카운터와 하나의 주기카운터를 사용하며, 코어에서 70개 이상의 이벤트를 모니터 할 수 있다. 카운터 중 하나에 오버플로우가 발생하여 생성된 인터럽트를 지시하여 소프트웨어 루틴의 읽기와 레지스터의 연산을 할 수 있다. 성능 카운터와 성능 이벤트는 코어상의 코프로세서 14의 레지스터를 이용하여 접근 할 수 있다. 성능 모니터링 제어 레지스터(PMNC), 클럭 카운터(CCNT), 인터럽트 활성화 레지스터(INTEN), 오버플로우 플래그 상태 레지스터(FLAG), 이벤트 선택 레지스터(EVTSEL), 성능 카운트 레지스터(PMN0, PMN1, PMN2, PMN3)들을 이용하여 성능 모니터링을 할 수 있다.

3. Windows CE 환경에서의 DVFS와 성능 모니터링을 위한 드라이버 구현

본 논문에서 사용한 프로그램은 DVFS와 성능 모니터링 제어를 위한 디바이스 드라이버와 이것을 이용하는 응용 프로그램으로 구성되어 있다. Windows CE OS 환경에서 전압/주파수 변경과 성능 모니터링을 제어하기 위해서는 특권 모드가 필요하기 때문에 디바이스 드라이버를 제작하여 드라이버와의 응용 프로그램간의 통신을 통하여 성능 모니터링을 제어하였다. 그림 1은 논문에 사용된 프로그램의 구성도를 보여준다.



(그림 1) 프로그램 구성도

```
XScaleReadPMUReg            FUNCTION

    cmp     r0, #8
    addls   pc, pc, r0, lsl #2
    b                       RRet
    b                       RdPMNC
    b                       RdCCNT
    b                       RdPMN0
    b                       RdPMN1
    b                       RdPMN2
    b                       RdPMN3
    b                       RdINTEN
    b                       RdFLAG
    b                       RdEVTSEL

RdPMNC
    mrc     p14, 0, r0, c0, c1, 0
 PMNC
    b       RRet
RdCCNT
    mrc     p14, 0, r0, c1, c1, 0
CCNT
    b                       RRet
RdPMN0
    mrc     p14, 0, r0, c0, c2, 0
PMN0
    b                       RRet
RdPMN1
    mrc     p14, 0, r0, c1, c2, 0
PMN1
    b                       RRet
RdPMN2
    mrc     p14, 0, r0, c2, c2, 0
PMN2
    b                       RRet
RdPMN3
    mrc     p14, 0, r0, c3, c2, 0
PMN3
    b                       RRet
RdINTEN
    mrc     p14, 0, r0, c4, c1, 0
INTEN
    b                       RRet
RdFLAG
    mrc     p14, 0, r0, c5, c1, 0
FLAG
    b                       RRet
RdEVTSEL
    mrc     p14, 0, r0, c8, c1, 0
EVTSEL

RRet
            IF :DEF: Interworking
      IF Interworking :LOR: Thumbing
        bx   lr
      ELSE
        mov   pc, lr            ; return
      ENDIF
            ELSE
      mov   pc, lr              ; return
    ENDIF

    ENDFUNC
```

(그림 2) 성능 모니터링 레지스터 값을 읽는 어셈블리언어 소스 코드

전압/주파수를 함께 변경하기 위해서는 다음 절차를 따른다.

1. PVCR (Power Management Unit Voltage Change Control Register) 레지스터의 FVE (Frequency/Voltage Change Enable) 값을 1로 설정하여, 주파수/전압 변경을 활성화 시킨다.
2. ASCR (Application Subsystem Power Status / Configuration Register) 레지스터의 MTS (Maximum Turbo Setting) 값을 터보모드와 런 모드에 따라서 1이나 2로 설정한다.
3. ACCR (Application Subsystem Clock Configuration Register) 레지스터의 XL과 XN에 변경을 원하는 주파수 설정에 해당하는 값으로 설정한다.
4. XCLCFG (Core Clock Configuration Register) 레지스터의 F비트를 1로 설정함으로써 주파수 변경을 시작한다.

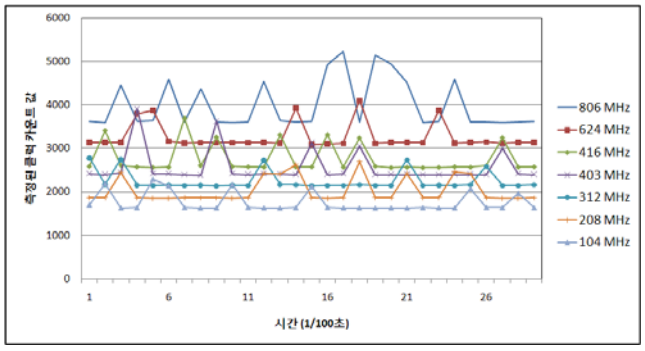
성능 모니터링 루틴은 4가지의 성능 이벤트 선택과 인터럽트 활성화/비활성화, 이벤트 카운터 활성화/비활성화를 설정하는 함수들로 구성 되어있다. 클럭 카운터, 성능 모니터 카운트 레지스터를 초기화하고, 4개의 성능 모니터 카운트 레지스터에 해당 이벤트를 설정한다. 그 후에 INTEN 레지스터를 활성화 시키고 카운트를 시작하여 성능 이벤트 값을 측정 할 수 있다. 그림 2는 코프로세서14의 성능 모니터링 레지스터 값을 읽는 어셈블리 언어 소스 코드이다. XScaleReadPMUReg 함수는 매개변수로 받은 성능 모니터링 레지스터에 대해서 그 값을 읽어내는 함수이다.

위와 같이 구현된 프로그램은 응용 프로그램에서 드라이버와의 통신을 통하여 전압/주파수 변경과 성능 모니터링을 제어한다.

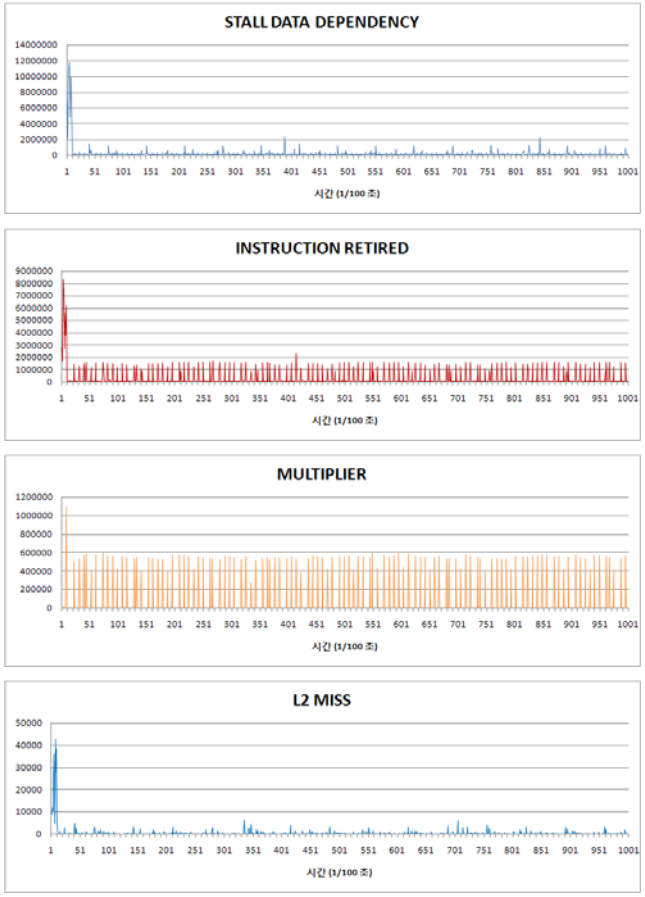
4. 구현 및 평가

성능 모니터링 측정에 있어서, 모든 성능 이벤트가 전력 소모량에 영향을 주지는 않는다. 그렇기 때문에 전력 소모량에 영향을 주는 성능 이벤트들 선택하여야 한다. PXA320 프로세서는 4가지 성능 이벤트만을 동시에 측정 가능하기 때문에 다음과 같은 4가지 성능 이벤트를 선택하였다. : Stall Data Dependency, Instruction Retired, Multiplier, L2 Cache Miss

4가지의 성능 이벤트를 설정하고, 7가지의 전압/주파수 구성에 따라서 매 10ms 마다 성능 이벤트 값과 클럭 카운트 값을 미디어 플레이어 프로그램에 대하여 측정하였다. 그림 3은 7가지의 전압/주파수 구성에 따른 클럭 카운트 값의 차이를 보여준다. 카운트 값은 코어 주파수가 높아지면 증가하였고, 낮아지면 감소하였다. 카운트 값이 감소하면 그만큼 전력 소모량도 감소하지만, 성능은 떨어진다. 전력 소모량도 줄이면서 성능에 크게 영향을 미치지 않는 전압/주파수 변경 지점을 찾아서 DVFS 기법의 스케줄링에 적용하면 전력 소모량 감소에 도움이 된다.



(그림 3) 미디어 플레이어 프로그램에 대하여 각각의 CPU 주파수에서 측정된 클럭 카운트 값



(그림 4) 미디어 플레이어 프로그램에 대하여 806MHz에서 측정된 4가지 성능 이벤트 값

그림 4는 미디어 플레이어 프로그램에 대하여 806MHz에서 측정된 4가지 성능 이벤트 값을 보여준다. 프로그램 로딩에서부터 10초간 측정되었고, 로딩 되는 시점에서 높은 측정값을 보이지만 이후로는 일정한 간격을 유지하는 결과를 볼 수 있다.

5. 결론

본 논문은 Windows CE 운영체제 환경에서 Intel XScale 마이크로아키텍쳐 기반의 Marvell PXA320 프로세서의 DVFS 기법을 위한 성능 모니터링을 구현하였다.

전력 소모량에 영향을 주는 4가지 성능 이벤트를 선택하였고, 각각의 코어 전압/주파수에 대하여 성능 이벤트 값과 클럭 카운트 값을 측정 하였다. 성능 모니터링으로 측정된 값을 기반으로 DVFS 기법의 스케줄링에 필요한 전압/주파수 변경 지점을 예측이 가능하다. 이것은 향후에 전력 소모량을 줄이는 다양한 전력관리 기법에 이용할 수 있다.

참고문헌

[1] Monahans P Processor - System and Timer Configuration Developers Manual, Vol. I

[2] Marvell PXA320 Processor - Electrical, Mechanical, and Thermal Specification

[3] Intel XScale® Core - Developer's Manual

[4] Performance Profiling Techniques on Intel® XScale TM Microarchitecture Processors

[5] 3rd Generation Intel XScale® Microarchitecture - Developer's Manual

[6] ARM Architecture Reference Manual

[7] Microsoft MSDN Library

[8] Glberto Contreras, Margaret Martonosi "Power Prediction for Intel XScale Processors Using Performance Monitoring Unit Events", International Symposium on Low Power Electronics and Design (ISLPED'05), August 2005

[9] Cantu가 Isci, Gilberto Contreras and Margaret martonosi "Live, Runtime Phase Monitoring and Prediction on Real Systems with Application to Dynamic Power Management", In 39th ACM/IEEE International Symposium on Microarchitecture (MICRO-39), Orlano, FL, Dec. 2006

[10] K. Rajamani, H. Hanson, J. Rubio, S. Ghiasi and F. Rawson, "Application-Aware Power Management", 2006 IEE International Symposium on Workload Characterization (IISWC-2006), Oct. 2006

[11] C. Poellabauer, T. Zhang, S. Pande and K. Schwan, "An Efficient Frequecy Scaling Approach for Energy-Aware Embedded Real-Time Systems", Preceedings of the International Conference on Architecture of computing Systems (ARCS'05), March 2005

[12] Le Yan, Lin Zhong, and Niraj K. Jha "User-perceived latency based dynamic voltage scaling for interactive applications" Proc. ACM/IEEE Desing Automation conf. (DAC), June, 2005