

하드웨어 성능 카운터를 이용한 임베디드 멀티프로세서

환경에서의 에너지 모델링과 예측 기법

최원일*, 김현희, 김지홍

서울대학교 컴퓨터공학부

{choi11, hh0726, jihong}@davinci.snu.ac.kr

Energy Modeling and Prediction Techniques for Embedded Multiprocessors Using Hardware Performance Counters

Wonil Choi*, Hyunhee Kim, jihong Kim

School of Computer Science and Engineering, Seoul National University

요 약

배터리 기반의 모바일 임베디드 시스템에서는 가용 에너지의 제약이 있기 때문에 프로그램 개발자들은 프로그램이 프로세서에서 적은 에너지를 소모하도록 최적화한다. 그러나 프로세서의 소모 에너지를 알기 위해서는 전력 소모를 측정할 수 있는 특별한 기기의 도움이 필요하다. 이것은 비용과 시간에 있어서 큰 제약이다. 한편, 최근에는 임베디드 시스템에서 멀티프로세서를 채택하는 비율이 높아지고 있다. 멀티프로세서는 에너지 소모 패턴이 다양하기 때문에 에너지를 관리하기 위한 에너지 관찰이 더욱 필요하다. 본 논문에서는 하드웨어 성능 카운터(Hardware Performance Counter)의 이벤트 값들과 선형 회귀 식을 통해서 전력 측정 도구 없이 멀티프로세서의 소모 에너지를 유추하는 방법론을 제공한다. 특히, ARM11 MPCore의 환경에서 에너지 모델을 만들어 보고 정확성을 검증한다. SPALSH-2 프로그램들에 대해서 최대 5% 오차 이내의 정확성을 보여준다.

1. 서론

임베디드 시스템에서 수행되는 프로그램을 만드는 개발자는 빠른 성능이나 적은 에너지 소모 등 다양한 최적화 목적에 따라 프로그램을 개선시킨다. 특히, 배터리 기반의 모바일 임베디드 시스템에서는 사용할 수 있는 에너지가 한정되어 있기 때문에 효과적인 에너지 사용이 절대적으로 요구되면서 에너지 최적화의 중요성이 점점 부각되고 있다.

개발자가 프로그램의 에너지를 최적화하기 위해서는 개발한 프로그램의 에너지 소모량을 알아야 한다. 소모 에너지를 알기 위해서는 전력 소모를 측정할 수 있는 특별한 기기의 도움이 필요하다[6]. 전력 측정 도구의 도움을 받는 것은 비용의 제약이 크다. 또, 프로그램의 성능 측정 또는 최적화와는 별개로 이루어져야 하기 때문에 측정 및 최적화의 시간이 더 요구된다. 이러한 전력 측정 도구의 제약 때문에

성능 정보를 이용하여 에너지 정보를 유추하는 방법들이 제안되었다. 다양한 프로세서들에서 하드웨어 성능 카운터(Hardware Performance Counter, 이하 HPC)들이 제공되는데, 프로그램 수행 동안 얻어진 성능 정보들을 사용하여 에너지를 예측하는 것이다.

한편, 최근에는 임베디드 시스템에서 멀티프로세서를 채택하는 비율이 높아지고 있다. 단일 프로세서의 속도, 전력, 발열 등의 문제를 해결하기 위해 복수 개의 프로세서를 하나의 칩으로 집약하기 시작했다. 그러나 멀티프로세서에서는 각각의 프로세서가 항상 사용되는 것이 아니기 때문에 에너지 소모 패턴의 변화가 크다. 따라서 멀티프로세서에서는 필수적으로 소모 에너지의 관찰과 관리가 요구된다. 단일 프로세서와 마찬가지로, 멀티프로세서도 에너지 소모를 알기 위해서 제약이 많은 전력 측정 도구를 사용하는 것 보다 HPC를 통해서 수집된 성능 정보들을 사용하여 에너지 정보를 예측하는 실용적인 방법이 요구된다.

본 논문에서는 멀티프로세서에서 1) 에너지 소모와 상관 관계가 큰 성능 정보들을 찾아내고, 2) HPC들을 통해서 성능 정보들을 수집하고, 전력 측정 도구를 통해서 전력 정보들을 수집한다. 수집된 성능, 전력 정보들은 3) 회귀 분석을 통해서 선형 식의 에너지 모델을 만드는데 사용된다. 또, 다양한 벤치마크 프로그램을 사용하여 4) 에너지 모델을 검증한다. 2)의 과정에서, 전력 정보를 수집하는데 있어서 전력 측정 도구가 사용된다. 그러나 에너지 모델이 완성되면, 전력 측정 도구 없이 성능 정보들만으로 에너지 소모를 예측할 수 있다. 특히, 사례 연구로써 ARM11 MPCore의 환경에서, 에너지 모델을 통하여 에너지 소모를 예측했으며, SPLASH-2 프로그램들에 대해서 최대 5%, 평균 3% 이내의 오차를 나타냈다.

이후의 논문 구성은 다음과 같다. 2장에서는 관련 연구를 소개하며, 3장에서는 에너지 예측 모델을 사용해서 멀티프로세서의 에너지 소모량을 유추하는 방법론을 설명한다. 4장에서는 ARM11 MPCore에서 에너지 예측 모델을 만들어 보고, 5장에서 실험 결과를 통해 ARM11 MPCore용 에너지 모델의 정확성을 분석한다. 이후 6장에서는 본 연구의 결론을 내린다.

2. 관련 연구

현재까지 제안된, 프로그램이 프로세서에서 소모한 에너지를 측정할 수 있는 기법은 시뮬레이션을 통한 기법[7], 실측을 통한 기법[3,6]과 성능 정보를 이용해 예측하는 기법[1,2,8]으로 분류된다.

SimplePower[7]는 합리적인 시간에 에너지 소모량을 추정하는 시뮬레이션 기법이다. 그러나 정확성이 낮기 때문에 개발자들이 프로그램을 분석하고 최적화하는데 실용적이지 못하다.

PowerScope[3]과 ePRO[6]은 전력 측정 도구를 사용하여 프로그램의 에너지 소모를 실측하는 기법이다. PowerScope은 에너지 정보만, ePRO는 성능 정보와 에너지 정보를 함께 제공한다. 비교적 정확하게 에너지 소모를 측정하지만, 전력 측정 도구의 도움이 필요하기 때문에 제약이 많다.

Contreras[1]과 Yun[8]은 Intel XScale 프로세서를 기반으로 성능 정보로 소모 전력을 예측하는 전력 모델을 제안하였다. 프로그램을 수행하면서 일정 구간에서 얻을 수 있는 다섯 종류의 성능 정보들과 전력 정보를 이용하여 선형 회귀 분석을 통해 전력 모델을 만든다. 그러나 단일 프로세서를 타겟으로 한 에너지 모델이기 때문에 멀티프로세서에서의 사용은 어렵다. 한편, Blume[2]은 멀티프로세서에서의 에너지 모델을 제시했는데 본 논문에서 사례로 제시하는 환경과 같은 ARM11 MPCore를 사용한다. 그러나 명령어 별 소모 전력과 같은 정보를 별도로 측정해야 하는 등의 에너지 모델의 도출 과정이 복잡하다.

본 논문에서는 전체 프로세서의 전력이 측정 가능한 멀티프로세서에 대해서, 에너지에 영향을 끼치는 성능 정보들을 분류하고, 간단한 선형 회귀 분석을 통해서 에너지

예측 모델을 도출함으로써, 변형이 많은 멀티프로세서에 대해서 확장 가능한 방법론을 제시하고 있다.

3. 멀티프로세서용 에너지 예측 모델

3.1 에너지 모델을 위한 성능 정보 선택

프로세서 별로 제공되는 HPC들은 다양하며 그 수가 많다. 그러나 모든 종류의 성능 정보가 프로세서의 소모 에너지에 영향을 주는 것이 아니다. 따라서 많은 성능 정보 가운데서 에너지 모델의 변수로 사용될 성능 정보를 선택하는 것이 필요하다. 본 논문에서는, 다양한 성능 정보들과 소모 에너지의 상관 관계를 모르는 상황에서, HPC들에서 제공되는 대부분의 성능 정보를 모두 수집했다. 그러나 수집한 성능 정보와 전력 정보를 회귀 분석하는 과정에서 상관 관계가 없거나 부족한 성능 정보들을 제거함으로써 아래의 다섯 개의 성능 정보를 선택했다. ()속의 약어는 식 (1)에서 사용한다.

Instructions (*Instr*): 많은 명령어를 수행할수록 더 많은 수행 시간을 필요로 하며 더 많은 에너지를 소모하는 것을 관찰할 수 있다.

L1 Data Cache Accesses (*DL1Access*): L1캐시는 프로세서가 가장 빈번하게 접근하는 최상위 메모리 계층이다. 특히 데이터 L1캐시는 명령어 L1캐시보다 접근 빈도가 높다.

L2 Cache Accesses (*L2Access*): L2캐시로 접근하는데 있어서 L1캐시보다 더 많은 시간이 필요하기 때문에 더 많은 에너지가 소모된다.

Data Dependencies (*DataDep*): 파이프라인의 데이터 의존성 문제를 해결하기 위해서 각각의 프로세서는 쉬는 경우가 많은데 이 때, 적은 전력을 소모하게 되며, 처리 시간 때문에 에너지 소모는 늘어나게 된다.

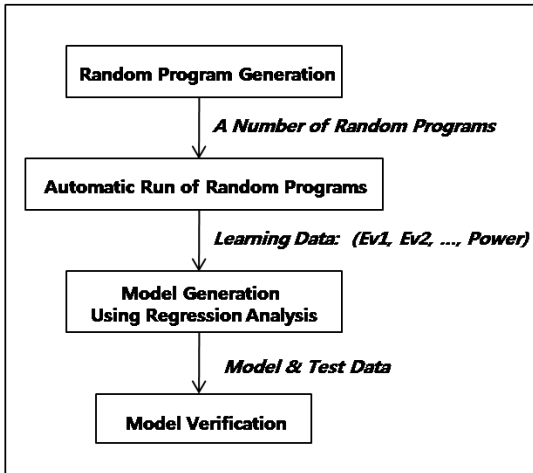
Coherence Transactions (*CohTrans*): 단일 프로세서와 달리 멀티프로세서에서는 프로세서간의 동기화 문제 때문에 프로세서 간의 빈번한 접근이 이루어진다.

위의 다섯 가지 성능 정보들과 멀티프로세서의 에너지 소모는 선형적인 관계를 가지고 있다고 가정하면 에너지 모델은 다음과 같다. 단위 시간당 성능 정보를 사용하여 단위 시간당 소모 전력을 구한 다음 전체 수행 시간을 곱함으로써 전체 소모 에너지를 알 수 있다.

$$Power = A*(Instr / time) + B*(DL1Access / time) + C*(L2Access / time) + D*(DataDep / time) + E*(CohTrans / time) + F_{const}. \quad (1)$$

$$Energy = Power * time \quad (2)$$

A~F를 많은 양의 성능 정보와 전력 정보의 조합으로 이루어진 데이터들의 회귀 분석을 통해 결정함으로써 전력 예측 모델 식 (1)을 완성할 수 있다. 전력 모델을 통해서 얻은 평균 소모 전력 값에 시간으로 곱하여 (식 2) 소모 에너지를 얻을 수 있다.



(그림 1) 에너지 모델의 도출 순서도

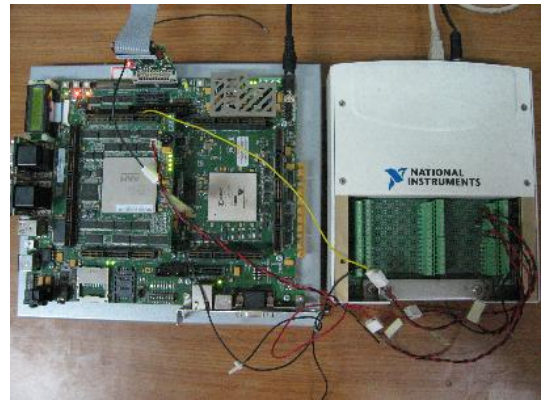
3.2 에너지 모델의 도출 과정

본 절에서는 성능 정보와 전력 정보로 이루어진 많은 데이터들을 회귀 분석하고 이를 이용하여 임의의 프로그램에 대해서 수집된 성능 정보들을 이용해서 소모 에너지를 예측할 수 있는 멀티프로세서용 전력, 에너지 모델을 만드는 과정을 설명한다. 그림 1은 에너지 모델을 만드는 전체 과정을 보여준다.

Random Program Generation: 에너지 예측 모델은 임의의 프로그램에 대해서 수집한 성능 정보들로 소모 에너지를 예측할 수 있어야 한다. 정확한 예측을 하기 위해 다양한 특성을 갖는 프로그램들의 성능 정보와 소모 전력 정보 조합의 데이터들이 요구된다. 본 논문에서는 다양한 패턴의 많은 프로그램을 생성하기 위해서 랜덤 프로그램 생성기를 만들어 사용한다. 예를 들어, 다양한 패턴의 함수들을 만들어 놓고 그 함수들의 임의의 조합을 호출하도록 프로그램들을 생성할 수 있다.

Automatic Run of Random Programs: 랜덤 프로그램 생성기를 통해서 만들어진 많은 양의 프로그램을 통해서 회귀 분석에 사용할 학습 데이터를 만드는 과정이다. 학습 데이터는 하나의 프로그램을 수행 시키면서 얻을 수 있는 3.1절에서 설명한 다섯 개의 성능 정보와 에너지 정보로 이루어 지는데, 다섯 개의 성능 정보는 HPC들로 얻을 수 있으며, 에너지 정보는 전력 측정 도구가 필요하다. 수행해야 할 프로그램의 수가 많기 때문에 이 작업을 자동화함으로써 많은 양의 학습 데이터를 수집할 수 있다.

Model Generation Using Regression Analysis: 수집된 많은 양의 학습 데이터를 가지고 회귀 분석을 통해서 에너지 모델을 완성하는 단계이다. 결정 트리, 인공 신경 망 등 사용할 수 있는 예측 모델이 많지만 가장 간단하고 직관적인 선형 회귀 식으로도 충분히 예측 능력을 높일 수 있다. 데이터 학습과 모델 생성에는 다양한 통계 프로그램을 사용이 가능하다.



(그림 2) ARM11 MPCore 와 NI 전력 측정기

Model Verification: 만들어진 에너지 모델을 검증하는 과정이다. 학습에서 사용된 데이터 뿐 만 아니라 임의의 프로그램에 대해서도 예측의 정확성과 오차 정도를 확인해야 한다. 만약 임의의 프로그램에 대해서 예측 능력이 떨어진다면 해당 프로그램의 패턴을 분석하여 비슷한 프로그램을 포함한 학습 데이터를 가지고 다시 모델을 만드는 작업이 진행된다.

4. ARM11 MPCore의 에너지 예측 모델

본 장에서는 사례 연구로써, 이전 장에서 소개한 에너지 모델을 만드는 순서에 따라서 칩 멀티프로세서 가운데 하나인 ARM11 MPCore 환경에서 에너지 모델을 만들어 에너지 모델의 정확성을 검증하였다.

그림 2의 ARM11 MPCore[4]는 칩 멀티프로세서로서 네 개의 코어로 이루어져 있다. 온-칩(on-chip) L1캐시, L2캐시와 오프-칩(off-chip) 메모리가 있다. 그 밖에 자세한 구성은 표 1을 참고한다.

운영 체제는 KISOS라는 SMP용 실시간 운영 체제를 사용하였다. KISOS는 간단하고 가벼운 pthread의 구현을 제공하며 어떤 종류의 프로그램에 대해서도 SMP 환경에서의 수행이 가능하다.

전력 측정기는 National Instrument사의 DAQPad-6016(그림 2)를 사용한다. 많은 수의 채널과 높은 샘플 주기를 가지고 있으며 연동된 프로그램을 통해서 매 주기마다 각종 정보를 기록할 수 있다.

(표 1) ARM11 MPCore의 프로세서와 메모리 구성

프로세서	<ul style="list-style-type: none"> · 4개의 ARM11 processors · 각각 210 MHz
메모리 시스템	<ul style="list-style-type: none"> · 32KB 명령어 & 데이터 L1 개별 캐시 · 1MB L2 공유 캐시 · Off-Chip 메모리 · Snoop Control Unit (SCU)

위와 같은 환경에서 그림 1의 순서도에 따라서 ARM11 MPCore용 에너지 모델을 만든다. Random Program Generation 과정에서는, 30개의 다양한 패턴을 가지는 함수들을 임의로 조합하여 프로그램을 만드는 랜덤 프로그램 생성기를 구현하여, 임의의 프로그램 200여 개를 생성한다. Automatic Run of Random Programs 단계에서는, 자동으로 임의의 프로그램들을 수행하고 성능 및 에너지 정보를 수집하는 작업을 스크립트를 통하여 자동화한다. Model Generation Using Regression Analysis 단계에서는 이전 단계에서 수집한 다량의 학습 데이터들을 사용하여 회귀 분석을 실시한다. 모델링에 사용한 SAS9 Enterprise Miner는 다양한 분석을 제공한다. Model Verification에서 간단한 검증을 마친 ARM11 MPCore용 에너지 모델의 계수와 상수는 표 2에서 볼 수 있다. 표 2의 값들은 식 1의 전력 모델에서 사용된다.

(표 2) ARM11 MPCore용 전력 모델의 계수 값

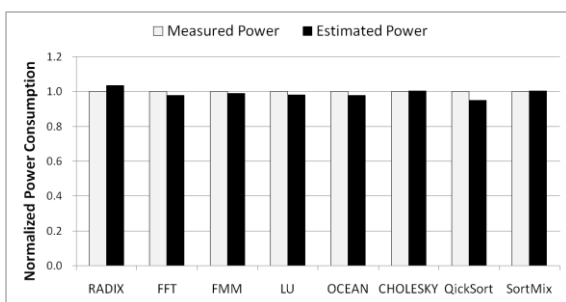
성능 정보 (계수)	값	성능 정보 (변수)	값
<i>Instr.</i> (A)	2.56E-07	<i>DataDep.</i> (D)	-3.97E-07
<i>DLIAccess</i> (B)	9.32E-07	<i>CohTrans</i> (E)	-1.99E-07
<i>L2Access</i> (C)	2.76E-05	<i>Const.</i> (F)	7.89E+02

5. 실험 결과

이전 장의 사례 연구 결과, 본 논문에서 제시한 멀티프로세서의 모델링을 통한 에너지 소모 예측을 위한 방법론을 검증할 수 있었다.

4장에서 도출한 ARM11 MPCore의 에너지 모델의 계수 값들인 표 2의 값들을 검증하기 위해서, 학습에서 사용한 랜덤 프로그램 이외에 벤치마크 프로그램으로 SPLASH-2[5]와 몇 개의 정렬 프로그램을 사용하여, 에너지 모델로 예측된 에너지 값과 전력 측정 도구를 사용하여 실측된 에너지의 소모를 비교하였다. 검증하고자 하는 멀티프로세서는 4개의 프로세서로 이루어져 있기 때문에 길이가 비슷한 네 개의 스레드를 생성하여 수행하였다.

그림 3은 여섯 개의 SPLASH-2과 Quick정렬 프로그램에 비롯한 몇 개의 정렬 프로그램에 대하여 성능 정보를 통해 추론한 평균 소모 전력과 NI 전력 측정 도구를 사용하여 실측한 평균 소모 전력을 비교한 그래프이다.



(그림 3) 측정 전력과 예측 전력의 비교

각각의 벤치마크 프로그램에 대해서는 다양한 input을 사용하여 비교해 보았다. 최대 5%의 오차를 보였으며 평균적으로는 3%의 오차를 나타냈다. 물론, 회귀 분석 시 학습하던 200여 개의 랜덤 프로그램에 대해서도 5%이내의 오차를 보였다. 평균 소모 전력 값에 시간을 곱함으로써 얻어지는 전체 에너지 값 역시 5% 이내의 오차를 보인다.

6. 결론

본 논문에서는 전력 측정 기기 없이 멀티프로세서에서 프로그램의 소모 에너지를 예측할 수 있는 방법론을 제시하였다. 멀티프로세서의 에너지 소모에 영향을 주는 다섯 개의 성능 정보를 밝히고 선형 회귀 모델을 세웠다. 많은 양의 랜덤 프로그램을 생성하고 학습 데이터를 만들어 모델의 예측의 정확성을 높이고, 자동으로 성능, 에너지 조합의 데이터를 수집함으로써 절차의 편리성을 높였다. 사례 연구로써, ARM11 MPCore 환경에서 에너지 모델을 만들었으며, SPLASH-2 프로그램들에 대해서 오차가 최대 5% 이내임을 보임으로써 정확성을 검증했다.

감사의 글

이 연구를 위해 연구장비를 지원하고 공간을 제공한 서울대학교 컴퓨터연구소에 감사 드립니다. 이 논문은 2008년도 두뇌한국21 사업과 삼성전자 시스템 LSI 사업부에 의해 지원되었으며, 2008년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구(No. ROA-2007-000-20116-0)입니다.

참고 문헌

- [1] G. Contreras and M. Martonosi, "Power Prediction for Intel XScale Processors Using Performance Monitoring Unit Events", in Proc. of ISLPED, 2005.
- [2] H. Blume, J. V. Livonius, L. Rotenberg, T. G. Noll, H. Bothe, and J. Brakensiek, "Performance and Power Analysis of Parallelized Implementations on an MPCore Multiprocessor Platform," in Proc. of IC-SAMOS, 2007.
- [3] J. Flinn and M. Satyanarayanan, "PowerScope: A Tool for Profiling the Energy Usage of Mobile Applications," in Proc. WMCSA, 1999.
- [4] ARM11 MPCore. [Online]. Available: <http://www.arm.com/products/CPUs/ARM11MPCoreMultiprocessor.html>
- [5] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 Programs: Characterization and Methodological Considerations," in Proc. of ISCA, 2005.
- [6] W. Baek, Y. Kim, and j. Kim, "ePRO: A Tool for Energy and Performance Profiling for Embedded Applications," in Proc. of ISOCC, 2004.
- [7] W. Ye, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "The Design and Use of SimplePower: A Cycle Accurate Energy Estimation Tool," in Proc. of DAC, 2000.
- [8] H. Yun, J. Sim, and S. Lee, "Power Prediction Using Performance Monitoring Counter in Mobile Processor," in Proc. of KCC, 2008.