

플랫폼을 기반으로 하는 SoC 설계 방법

최규명, 정의영, 엄준형, 어수관

삼성전자 System-LSI사업부

요 약

SoC(System on Chip)의 발전은 현재 널리 쓰이는 기존의 설계 방법론으로는 해결될 수 없는 여러 가지 문제들을 생성하고 있다. 제품의 복잡도는 급격히 증가하는 반면 제품 수명은 지속적으로 감소하고 있으며, 기존 구현 기술의 향상만으로는 해결하기 쉽지 않은 VDSM(Very Deep Sub-Micron) 문제 또한 새로이 도출되고 있는 실정이다. 이러한 문제들은 적기에 시장에 제품을 출시하는 데에 많은 어려움을 끼치게 되기 때문에, 기존의 방법론을 뛰어 넘는 새로운 설계 방법론을 위한 많은 연구들이 활발히 진행되고 있다. 이러한 방법론 중 효율적인 한 방안으로 “재사용”의 개념이 대두되고 있으며, platform-based design은 이러한 재사용의 개념을 SoC 설계에 효율적으로 적용할 수 있는 방법으로 많은 관심을 받고 있다. 본 논문은 이러한 platform-based design의 최근 동향 및 현재 삼성전자에서 추진하고 있는 platform-based design에 대해 기술한다.

I. 서 론

최근 전자 제품의 특징을 살펴보면 소형화, 복합화, 무선화로 요약될 수 있다. 소형화를 통한 휴대용 제품의 출시가 가능하고, 복합화를 통하여 하나의 제품이 다양한 기능을 제공하므로, 다

수의 제품을 휴대해야 하는 번거로움이 줄어들고 있다. 또한 이러한 휴대용 제품들은 무선 통신을 통한 정보 교환이 가능하므로 운용상의 용이함마저 제공한다. PDA, 전화, GPS 기능 등을 제공하며 휴대가 간편한 스마트 폰(smart phone)이 그 좋은 예라고 할 수 있으며, 이러한 제품들의 출현은 하나의 시스템이 하나의 칩상에 구현되는 SoC(System-on-chip) 설계의 보편화로 인해 가능하다.

근본적으로 SoC 설계는 실리콘 구현 기술의 급격한 발전으로 인한, 기존의 보드 레벨에서나 구현이 가능한 제품을 하나의 실리콘 상에 구현이 가능하게 되었으며, 이러한 집적도의 향상은 앞으로 더욱 가속화 될 전망이다. 이미 90nm 공정 기술을 이용한 제품이 출시되고 있으며, ITRS(International Technology Roadmap for Semiconductors)에 따르면, 2010년 경에는 50nm 공정 기술을 바탕으로, 1V 이하의 동작 전압에서 10GHz의 동작 속도로 구동되는 40억 트랜지스터가 집적된 회로의 설계가 가능할 것으로 예측되고 있다.

SoC 설계의 궁극적 목표는 개발 기간 및 비용의 관점에서 성능 및 기능 제약 조건(constraints)과 생산 비용을 효율적으로 절충(trade-off)하는 것으로 정의될 수 있다^[1]. 생산 비용의 심각성을 잘 나타내주는 한 예로 마스크(mask) 제작 비용을 들 수 있다. 130nm 공정에서 한 세트(set)의 마스크 제작 비용이 60만불 이상이 소요되는 것이 드문 일이 아니며, 차세대 공정의 제작 비용은 100만불 이상이 소요될 것으로 예측되고 있다^[1,2]. 또한, 130nm 또는 100nm 공정을 사

용하여 300 mm 웨이퍼(wafer)를 생산하는 대량 생산 공장을 짓기 위해서는 약 35억불의 예산이 소요될 것으로 추정되고 있다^[3]. 반면에 개발기간의 단축의 심각성은 갈수록 짧아지는 제품수명에서 그 예를 찾아볼 수 있다. 휴대 전화기(mobile phone)의 경우, TV광고를 통해서도 얼마나 새로운 제품이 빨리 출시되고 있는지를 쉽게 알 수 있다. 실리콘 구현 기술의 발전은 소형화 및 복잡화를 추구할 수 있는 기반 기술(infrastructure)을 제공하지만, 결국 이를 바탕으로 기능을 구현하는 설계자에게는 time-to-market의 압력이 갈수록 증가하고 있는 추세이다.

이러한 문제를 해결하기 위한 여러 가지 설계 기법들이 연구, 적용되고 있으며, 최근에는 platform-based design 설계 기법이 이러한 문제를 위한 효율적인 방안으로 제시되고 있다. 이 기법을 특징지으며 또한 가장 큰 장점이 되는 세가지 요소는 재사용(reuse), 유연성(flexibility) 및 효율성(efficiency)이며^[4], 이 중 재사용이란, 주어진 응용 분야(application domain)에 공통적으로 적용될 수 있는 아키텍처를 정의하여 동일한 아키텍처를 다시 사용함을 의미한다.(기존의 ASIC 설계시의 재사용 개념은 주로 하드웨어 IP의 재사용을 의미하였으나, platform-based design은 SW를 포함한 architecture의 재사용을 추구한다.) 이러한 아키텍처의 재사용을 광범위한 영역에서 적용하기 위해서는 유연성(flexibility)이 필수이기 때문에, 이를 위해 프로그래머빌리티(programmability)가 높은 아키텍처 구조가 platform-based design에 적합하다고 할 수 있다. 즉, 바꾸어 말하면 아키텍처를 최대한 프로그래머블하게 만드는 방법론을 개발하는 것이 platform-based design을 용이하게 한다고 할 수 있다. 결국, 프로그래머블(programmable)한 아키텍처의 재사용을 통하여 효율적인 생산이 가능하고, 따라서 마스크 제작 비용을 크게 낮추는 효과를 얻을 수 있다.

일반적으로 아키텍처를 구성하는 요소는 계산 유닛(computation unit), 통신 유닛(commu-

nication unit), 및 저장 유닛(storage unit)으로 정의될 수 있다. 계산 유닛의 프로그래머빌리티는 SW를 기반으로 한 프로세서의 사용으로 향상시킬 수 있으며, FPGA(Field Programmable Gate Array) 등의 재구성이 가능한 하드웨어(reconfigurable HW resource)도 최근 각광 받고 있다. 통신 유닛의 경우 기존의 와이어(physical wire) 수준의 버스 중심의 설계 개념이 아닌, 소규모 네트워크(micro network) 개념과 재구성의 개념이 결합된 통신 유닛 구조가 대두되고 있다. 이러한 통신 아키텍처의 제안은 여러 가지 이유가 있지만, 그 중 하나가 유연성을 높이기 위한 것이다.

이러한 유연성은 최적화되지 않은 설계(sub-optimal design)를 유발하는 단점을 가지고 있지만, 이러한 단점은 platform-based design 방법론의 적용이 time-to-market을 만족함에 있어 주는 장점을 고려할 때 어느 정도 허용될 것으로 예측된다. 그럼에도 불구하고 유연성을 효과적으로 이용하기 위하여, 주어진 제약 조건(constraints)을 만족시키는 최적의 프로그래밍은 필수적이며, platform-based design 기법을 이용한 다양하게 프로그램된 아키텍처에 대한 평가가 한가지 방안이다.

II장에서는 현재 진행되고 있는 platform-based design 기법의 연구 동향에 대해 살펴보고, III장에서는 삼성전사에서 추진하고 있는 platform-based design 기법에 대해 설명하기로 한다. IV장에서는 삼성전자 내에서 적용된 platform-based design 기법의 적용 사례 및 향후 계획에 대해 알아보고, 마지막으로 V장에서 글을 맺고자 한다.

II. Platform-based design의 연구동향

II장에서는 해외 선진 연구기관 및 기업에서 진행되고 있는 현황 및 적용 사례에 대해 알아보기로 한다. II장에서 소개되는 내용 중 많은 부분

이 참고문헌에서 발췌된 것으로 더 상세한 내용을 원할 경우 참고 문헌의 참조를 권한다.

1. Philips Video Top^[4]

유럽 공동체에서는 산업체(Philips, Siemens, Cadence)와 대학(University of Paris Marie-Curie, Politecnico di Torino, University of Tubingen)이 공동으로 시스템 설계 및 검증 목적으로 한 COSY라는 프로젝트가 진행되었다. 이 프로젝트의 일환으로 platform-based design 설계 기법의 산업체 제품 적용 과제가 수행되었으며, 그 대상 제품은 Philips의 디지털 비디오 브로드캐스팅(broadcasting)을 주기능으로 하는 Video Top이었다. Video Top은 MPEG2video stream을 decoding한 후 선택적으로 줌(zoom) 또는 합성(composition)을 수행한다.

Video Top은 Kahn process model에서 비롯된 YAPI를 이용하여 62 프로세스로 구성된 2,5000 C/C++ 라인 규모로 모델링 되었다. MIPS PR39K, 1M SRAM을 기반으로 PSOS를 RTOS로 사용하는 아키텍처로부터 출발하여 성능 분석을 통한 아키텍처 탐색을 수행하였으며, 모두 SW로 구현된 경우보다 HW와 SW로 구성된 아키텍처의 경우 160배의 성능 향상을 보임을 찾아내었다. 이중 가장 중요한 HW 블록은 메모리 액세스(access)를 효율적으로 수행하기 위한 프레임 버퍼 메모리 관리자(frame buffer memory manager)였다. 성능 분석은 Cadence사의 VCC를 이용하였으며, Philips의 cycle-accurate 시뮬레이터인 TSS에 비해 5% 정도의 오차를 보일 만큼 정확히 모델링되었다. Philips는 검증된 platform-based design 기법을 차세대 주요 제품에 지속적으로 적용할 계획이다.

2. Magneti-Marell 자동차 엔진 제어^[4]

자동차 엔진 제어 시스템의 설계는 안전성이 매우 중요시 되므로 엄격한 제약 조건이 고려되어야 한다. 이 과제는 기존에 구현된 C 코드(약 135,000 줄)를 2개의 프로세서, 메모리, 입출력

서브시스템이 크로스바 스위치(crossbar switch)에 연결된 새로운 아키텍처에 맞추어 재설계하였으며, 재설계된 SW는 참고 문헌 [1]에 제안된 바와 같은 계층별 소프트웨어 구조(layered software structure)의 개념을 이용하였으므로 또 다른 아키텍처에 쉽게 맵핑(mapping)되도록 설계되었다. 성능 분석을 통한 아키텍처 탐색 과정에서는 3개의 프로세서 종류 중 최적의 프로세서를 탐색하는 작업이 포함되었으며, 각 프로세서 종류에 대해 2개의 다른 소프트웨어 분할(partition)이 평가되었다. 이와 동시에 입출력 시스템에 대해 전체를 소프트웨어로 구현된 경우, 일부 주변장치만이 하드웨어로 구현된 경우, 모든 주변장치가 하드웨어로 구현된 경우에 대해 성능 평가가 이루어졌다.

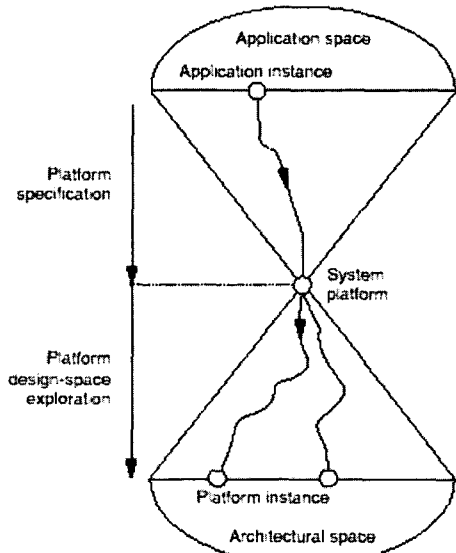
Cadence사의 VCC를 이용한 성능 분석은 프로토타입 보드(prototype board)에서의 측정 결과와 비교하여 약 11%의 오차를 보였다. 이중 하나의 아키텍처는 주어진 기능을 주어진 제약 조건 내에 수행하고도 약 40%의 계산능력(computing power)에 여유가 있음을 VCC를 통한 분석으로 알 수 있었다. 이러한 결과는 단일 프로세서만으로도 충분한 계산 능력을 제공될 수 있을 지에 대한 의문을 제기하지만, 전체 칩 면적의 대부분을 플래시 메모리(flash memory)가 차지함으로써, 두 번째 프로세서의 추가로 인한 면적 증가는 단지 4~6%에 불과하다는 점에서 2개 프로세서를 이용한 아키텍처가 적합하다고 할 수 있다.

한 가지 주목할 점은 Philips Video Top과 이 과제를 성공적으로 이끈 Cadence사의 VCC 툴이 더 이상 지원되지 않고 사라졌다는 것이다.

II. 삼성전자의 platform-based design 기법 구축 현황

1. 시스템 레벨 설계 플로우

<그림 1>은 platform-based design에 대한

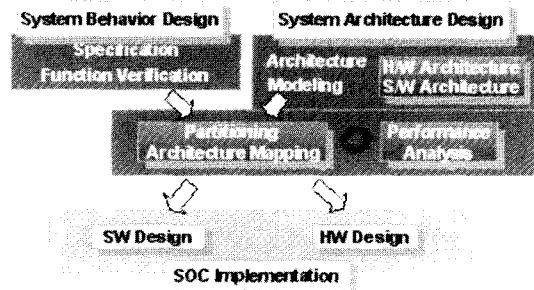


〈그림 1〉 플랫폼 추상화 및 설계 플로우^[4]

일반적인 방법론을 나타낸다.

위의 그림에서, platform-based design은 플랫폼 명세(platform specification) 단계와 플랫폼 아키텍처 탐색(platform architecture exploration) 단계로 구성되어 있다. 먼저, 시스템 설계자는 플랫폼 명세 단계에서 자신의 설계를 시스템 플랫폼에 매핑한다. 시스템 플랫폼(system platform)은 일종의 추상화된 하드웨어 플랫폼(hardware platform)을 의미한다. 이어 따르는 두 번째 단계인 플랫폼 아키텍처 탐색 단계에서 시스템 플랫폼은 일련의 실제 아키텍처 중에서 가장 최적화된 아키텍처를 선택하게 된다.

이 방법에서 가장 중요한 요소는 시스템 플랫폼의 추상화의 정도, 그리고 추상화 된 플랫폼으로부터 실제 아키텍처로 맵핑(mapping)시키는 일이다. 추상화 정도가 높으면 높을수록 시스템 설계자의 일은 쉬워지나, 이를 실제 아키텍처로 맵핑(mapping)하는 과정은 보다 많은 아키텍처에 대한 선택의 여지가 있게 되므로 그만큼 어려워지게 된다. 따라서, 이러한 맵핑(mapping) 과정은 자동화된 도구(software synthesis, RTOS synthesis, device-driver synthesis)의 도움



〈그림 2〉 삼성전자의 시스템 레벨 설계 플로우

이 필수적이다. 그러나 이러한 일반적인 방법론의 platform-based design은 아직 연구 단계이며, 상용화 단계에 이른 틀은 없는 실정이다. 그러므로, 삼성전자도 궁극적으로는 이와 같은 방법론을 추구할 예정이지만, 단기적 방안으로 〈그림 1〉과 같은 실질적인 해법을 강구하게 되었다.

〈그림 2〉는 삼성전자에서 추구하고 있는 시스템 레벨 설계 플로우(design flow)이다. 삼성전자에서는 2002년부터 platform-based design 기법에 대한 연구를 시작하였으며, 고부가가치 가정용 제품군과 휴대용 제품군을 대상으로 기술 개발을 시작하였다.

〈그림 2〉에서 우측 상단의 아키텍처 모델링 단계는 〈그림 1〉의 각 플랫폼 인스턴스(platform instance)들을 생성하는 과정이다. 이와 같이 준비된 각 플랫폼 인스턴스에 대해 주어진 명세(specification)를 맵핑하며 성능 측정을 수행하여 가장 적합한 아키텍처를 선택하게 된다. 이후 소프트웨어로 할당된 부분은 아키텍처 탐색 시 결정된 프로세서에 탑재되며, 하드웨어로 할당된 부분은 legacy IP가 바로 사용되거나 상위레벨 합성(high-level synthesis)을 통하여 생성된다. 현재 당사의 아키텍처 모델링 단계에서는 주로 legacy IP를 고려하여 아키텍처를 설계함으로써, 높은 재사용을 통한 설계 기간 단축을 추구한다.

2. 상위 단계 모델링

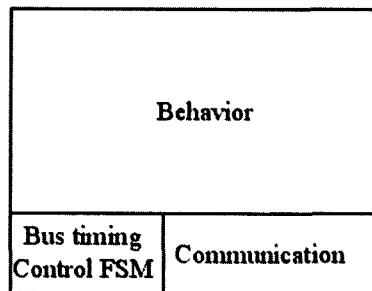
〈그림 2〉의 설계 플로우의 아키텍처 맵핑(mapping) 및 성능 분석 부분을 위해 당사에서

는 SystemC를 이용한 상위 단계 모델링을 이용하고 있다. 이에 대해 좀 더 자세히 기술하자면, 현재 platform-based design을 위해 사용하고 있는 모델링의 추상화 단계는 트랜잭션 단계(transaction-level)로써 이는 시뮬레이션을 통한 성능 분석 시간과 모델의 정확도를 절충하기 위한 모델링 기법이다. 이러한 방법론을 이용하여 추상화 된 모델들을 일반적으로 TLM(Transaction-Level Model)이라 한다.

〈그림 3〉에 당사의 있는 TLM 방식을 도식화하여 나타내었다.

〈그림 3〉에 나타낸 바와 같이 삼성의 TLM은 3개의 서브 블록(sub-block)으로 구성된다. 기능적 부분을 기술하는 behavior 블록, 통신망(communication network)과의 인터페이스를 기술하는 communication 블록, 그리고 버스 클럭 사이클 단계의 정확도를 유지하기 위한 bus timing control FSM 블록으로 구성된다. Behavior 블록은 untimed 또는 timed 단계에서 기술되므로 자세한 타이밍 정보의 필요 없이 알고리즘 수준에서 기술될 수 있다. 따라서 각 레지스터에서의 타이밍의 정확성이 요구되는 RTL 모델에 비해 월등한 시뮬레이션 속도의 향상을 가질 수 있다.

외부와의 데이터 통신은 communication 블록에 정의된 버퍼에 기록되며, communication 블록은 정해진 통신 프로토콜에 따라 버퍼에 저장된 통신 request를 외부 통신망에 요청하게 된다. 이때, request를 요청하는 타이밍을 bus timing control FSM 블록이 제어하게 되는데,



〈그림 3〉 SystemC를 이용한 TLM 구조

이 블록의 정확도가 전체 성능 측정의 정확도를 좌우하는 중요한 요소가 된다. Communication 블록과 통신망간의 인터페이스는 RTL 모델링에서 사용되는 pin-accurate한 신호 단계(signal-level)의 인터페이스가 아닌, 유관 신호들의 그룹으로 이루어진 transfer 또는 transfer의 그룹으로 이루어진 transaction 단계에서 이루어진다. 이상의 transaction 및 transfer는 통신망에서 정의된 함수들이며, 이들을 마스터 또는 슬레이브 모델에서 호출하여 통신이 이루어지는 방식이다. 따라서 각 신호의 이벤트를 일일이 처리해야 하는 RTL 방식에 비해 현격한 시뮬레이션 속도 향상을 갖는다.

비록 TLM은 모델 내부의 타이밍은 RTL 모델과 다르지만, 외부에서 보이는 timing behavior는 bus timing control FSM에 의해 제어되므로 RTL 모델과 매우 유사한(혹은 동일한) timing behavior를 가지게 된다. 이 블록은 RTL level의 legacy IP로부터 추출하게 되며, 현재는 수작업에 의해 구현되고 있다. 그러나, 일부 자동화가 용이한 특징을 갖는 IP에 대해서는 자동화를 위한 방법론을 개발, 구현 중에 있다. 새로운 IP의 경우에는 설계 진도에 따라 untimed model, timed model, TLM의 순으로 모델링 정확도를 높여가는 방식을 취하고 있다.

TLM은 RTL 모델과 유사한 정확도를 가지면서도 고속의 성능 분석이 가능하다는 큰 장점을 가지며, 이는 결국 다양한 아키텍처의 탐색이 가능하게 한다. 또한, 기존의 RTL legacy IP와 같이 재사용이 가능하며, communication 블록이 별도로 구분되어 있어 연결되는 통신망의 변화에 대해 손쉽게 인터페이스의 수정이 가능하다는 장점도 가지고 있다.

3. 프로세서 모델

소프트웨어의 비중이 높아짐에 따라 커스텀 IP의 상위레벨 모델과 더불어 중요한 것이 프로세서 모델이며, 프로세서 모델의 확보에도 많은 노력을 기울이고 있다. ARM과 같은 상용화된 프로세서 모델은 EDA 업체에서 공급하는 경우가

많지만, 커스텀 프로세서의 경우에는 이에 대한 모델의 확보가 필수적이다. 특히, 당사의 경우에는 하나의 프로세서에 대해 IA(instruction-accurate) 모델과 CA(cycle-accurate) 모델을 모두 준비하여, 타이밍이 중요하지 않은 성능 분석에서는 IA 모델을 (예: cache hit/miss ratio), 타이밍이 중요한 경우에는 CA 모델을 쓰는 이원화된 방법을 지원하고 있다. IA 모델 및 CA 모델 모두 TLM 인터페이스로 기술되어 통신망과의 인터페이스가 매우 용이하다.

4. Virtual Platform

TLM은 3.2절에서 언급된 바와 같은 장점 이외에도 빠른 시뮬레이션 속도를 바탕으로 소프트웨어를 개발할 수 있는 환경도 제공한다는 장점을 가지고 있다. 예를 들어, device driver는 RTL 설계가 완료된 후 FPGA prototyping을 통하여 마련된 보드상에서 진행되는 것이 전통적인 방법이나, TLM을 이용하여 시뮬레이션 모델을 소프트웨어 개발자에게 미리 제공할 경우 RTL 설계 이전에도 코드 개발이 가능하다. 또한 FPGA 보드상에서는 에러가 발생할 경우, 소프트웨어에 의한 에러인지, 하드웨어에 의한 에러인지를 규명하는 것이 간단하지가 않다. 그러나 제안된 환경에서는 HW/SW co-debugging이 가능하여, 에러의 원인을 찾아내는 일이 수월하다. 당사에서는 이러한 시뮬레이션 모델을 가상 플랫폼(ViP: Virtual Platform)이라 명명하였으며, 이러한 환경은 앞서 기술한 소프트웨어의 개발 및 HW/SW co-simulation을 통한 성능 분석을 가능하게 한다. ViP의 시뮬레이션 환경은 CoWare사의 ConvergenSC와 AxyS사의 Max-Sim을 이용하여 구축되어 있다.

ViP는 또한 RTL IP와의 co-simulation도 가능하므로, 개발된 IP를 시스템 레벨에서 검증할 수 있는 환경도 제공하고 있다. TLM 통신망과 RTL IP의 연결은 transaction을 신호로 (혹은 그 반대) 변환하는 어댑터(adaptor) 블록을 삽입하여 가능하다. 하나의 통신 프로토콜에 하나의 어댑터만을 준비하면 되므로, 어댑터 개

발은 크게 어렵지 않다.

또 다른 ViP의 장점은 반도체 회사가 칩이 완성되기 이전에, 시스템 업체에 시뮬레이션 모델을 제공할 수 있다는 것이다. 이 시뮬레이션 모델을 이용하여 시스템 업체는 칩이 공급되기 이전에 시스템 설계를 진행할 수 있다.

IV. 삼성전자의 Platform-based design 적용 사례

1. 적용 현황

현재 당사에서는 각 응용 분야별로 ViP를 준비하는 과정에 있다. 특히 가정용 제품 응용 분야(디지털 TV, set top box 등)와 휴대용 제품(스마트 폰)에 대해 역량을 집중하고 있다. 이들 응용 분야에 대한 ViP 구축에서 가장 많은 노력을 기울이고 있는 부분은 TLM 및 프로세서 모델 개발인데, 이는 앞서 언급하였듯이 개발된 후에 재사용이 가능하기 때문이다. 가정용 제품 응용 분야 및 휴대용 제품 응용 분야의 ViP는 2003년 말에 완성될 계획이며, 네트워크 분야의 ViP는 2003년 9월에 완성될 계획이다. III장에서 거론된 상위레벨 모델링 기법을 적용하지는 못했지만, RTC(Register Transfer-level C)로 모델링한 PDA ViP를 2002년도에 완성한 바 있다. 다음의 4.2절에서는 이 적용 사례를 구체적으로 기술한다.

2. Pele: PDA용 ViP 개발 프로젝트

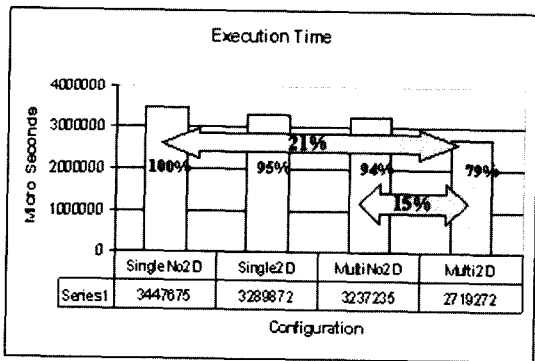
Pele 프로젝트의 대상 PDA 플랫폼은 단일 프로세서(ARM926EJS)를 기반으로 하며, 카메라 기능을 내장하였고, 그래픽 처리를 신속히 수행하기 위하여 2D 그래픽 가속기(graphics accelerator)를 내장하였다. 본 프로젝트에서는 다음의 4가지 아키텍처에 대해 성능 분석을 통한 아키텍처 탐색을 수행하였다:

1. 단일 버스 레이어와 SW 2D 그래픽 가속기
2. 다중 버스 레이어와 SW 2D 그래픽 가속기

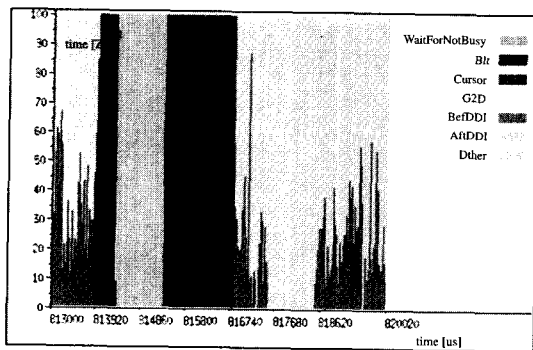
- 3. 단일 버스 레이어와 HW 2D 그래픽 가속기
- 4. 다중 버스 레이어와 HW 2D 그래픽 가속기

다중 버스 레이어를 사용할 경우에는 bus matrix 를 사용하여 메모리 액세스 충돌을 (memory access conflict) 해결하였으며, 앞서 제시한 4 가지 아키텍처에 대해 WinCE.net을 탑재하여 RTOS 효과까지 고려한 성능 분석이 이루어졌다. 이러한 성능 분석의 결과를 <그림 4>에 나타 내었다.

예측되었듯이 다중 버스 레이어와 HW 2D 그래픽 가속기를 사용했을 경우의 성능이 가장 좋은 것으로 평가되었다. 여기서 주목할 점은 성능 개선의 정도를 정량적으로 평가했다는 점이며, 이는 기존의 RTL 설계 기법으로서는 time-to-market을 고려할 때 거의 불가능한 비교이다. 버스 레이어 혹은 2D 그래픽 가속기 중 한가지 만 개선했을 경우에는 원하는 성능에 못 미친다



<그림 4> 4가지 아키텍처에 대한 성능 분석 결과



<그림 5> CPU 로드 분석(다중 레이어, HW 2D)

는 분석도 가능하였으며, <그림 5>에 보이는 CPU load 분석을 통하여, HW 2D 그래픽 가속기의 동작이 끝나기를 기다리며 검사하는 WaitForNotBusy 함수가 CPU를 점유하여, 다른 task의 수행이 지연되는 현상을 발견할 수 있었다. 이 결과로부터 2D 그래픽 가속기에 동작 완료를 인터럽트(interrupt) 방식으로 처리하는 개선안을 찾을 수 있었다. 이외에도 SW 분석을 통하여 계산량을 많이 요구하는 부분을 찾아낼 수 있었으며, 각 버스의 사용량(utilization)의 분석도 가능하였다.

IV. 결 론

본 논문에서는 SoC 설계를 효율적으로 수행할 수 있는 platform-based design 기법의 개념과 각 선진사의 적용 현황, 그리고 삼성전자에서 진행하고 있는 현황에 대해 살펴보았다. 삼성전자의 방법론은 궁극적으로 이상적인 플로우를 추구하지만, 단기적으로는 상위 레벨 모델링과 재사용의 개념을 적극 활용하는 실질적 방법론을 적용하고 있다. 이미 PDA용 제품에 적용한 사례를 통하여 platform-based design 기법의 효용성이 입증되었으며, 2003년도 말에는 추가적으로 타 제품들에 적용한 결과를 얻을 예정이다.

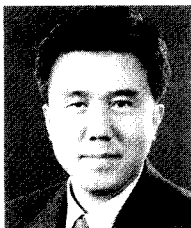
참 고 문 헌

- (1) A. Sangiovanni-Vincentelli and G. Martin, "Platform-Based Design and Software Design Methodology for Embedded Systems", *IEEE Design& Test of computers*, pp.23-33, vol. 18, no. 6, Nov. 2001
- (2) P. Magarshack, "Improving SoC Design Quality through a Reproducible Design Flow", *IEEE Design&Test of computers*,

pp. 76-83, vol. 19, no. 1, Jan. 2001
 [3] S. P. Dean, "ASIC and Foundaries", Integrated System Design, Mar. 2001 ; <http://www.isdmag.com/story>
 [4] K. Keutzer, S. Malik, R. Newton, J. Rabaey, and A. Sangiovanni-Vincen-

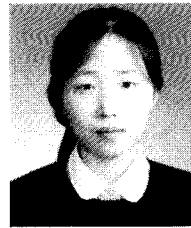
telli, "System Level Design : Orthogonalization of Concerns and Platform based design", *IEEE trans. on Computer-Aided Design*, pp.1523-1543, vol. 19, no. 12, Dec. 2000

저자 소개



최규명

1983년 2월 한양대학교 전자공학과 졸업, 1985년 2월 한양대학원 전자공학과 졸업 (MS), 1995년 8월 Univ. of Pittsburg, EE Dept. 졸업 (PhD), 1985년 1월~1991년 8월 : 삼성반도체통신, 삼성전자 CAE팀 연구원 (선임, 책임), 1995년 10월~2000년 11월 : 삼성전자 Memory사업부 CAE팀 책임/수석연구원, 2000년 12월~현재 : 삼성전자 System-LSI사업부 CAE Center 수석연구원, 그룹장, <주관심 분야 : Design Methodology for SOC and VDSM Design>



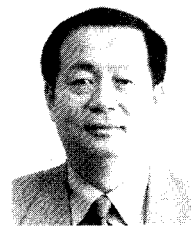
엄준형

1996년 2월 서울대학교 수학교육학과 졸업, 1999년 2월 한국과학기술원 수학과 졸업, 2003년 2월 한국과학기술원 전산학과 졸업 (PhD), 2003년 1월~현재 : 삼성전자 System-LSI사업부 CAE Center 책임연구원, <주관심 분야 : High-level synthesis considering layout effect, Embedded SW design and optimization, Low power crypto system design>



정의영

1988년 2월 고려대학교 전자공학과 졸업, 1990년 2월 고려대학원 전자공학과 졸업 (MS), 1997년 6월 Stanford university, EE Dept. 졸업 (PhD), 1990년 1월~1995년 12월 : 삼성반도체통신, 삼성전자 CAE팀 선임연구원, 1996년 1월~2003년 2월 : 삼성전자 Memory사업부 CAE팀 책임연구원, 2003년 3월~현재 : 삼성전자 System-LSI사업부 CAE Center 수석연구원, <주관심 분야 : SOC design methodology, low power design>



어수관

1975년 2월 서강대학교 물리학과 졸업, 1986년 6월 University of Arizona 전자공학과 졸업 (MS), 1986년 1월~2002년 5월 : 미국 Cadence, Synopsys, Intel, Via technology 등에서 재직, 2002년 5월~현재 : 삼성전자 System-LSI사업부 CAE Center 연구위원, <주관심 분야 : SOC design methodology>