

최근 시스템 설계 경향과 방법론

홍진석, 김주선, 배정환
삼성전자 중앙연구소 NS Lab.

전화:031-200-4769 / 이동전화:017-255-9732

The Current System Design Trends and Methodologies

Jin Seok Hong, Joo Sun Kim, Jum Han Bae

NS Lab., Corporate R/D Center, Samsung Electronics

{ emailed, snjskim, jhbae }@samsung.com

Abstract

With increasing design complexity, shrinking time-to-market window, and various demands from markets, the traditional design methodologies reached a ceiling. In a search for a way to tackle these problems, novel concepts and methodologies have been proposed in the past years. The paper briefly presents the design issues, and concepts, introduces commercial tools, and finally proposes a conceptual methodology that addresses the issues.

I. 서론

통상 디지털, 또는 아날로그 그리고 칩, 보드, 세트 등 다양한 관점으로 시스템을 이해하고 각각 나름대로 독자적인 개발방식을 적용시켜 왔다. 그러나 공정기술의 발전과 더불어 다양한 구성 요소들을(Logic, SRAM, Flash Memory, DRAM, CMOS RF, FPGA, MEMS, FRAM, Chemical Sensors, Electro-optical, Electro-biological Component) [1] 단일 칩에 이미 집적할 수 있거나 조만간 가능하게 될 전망이어서 SoC (System On a Chip), IAoC(Information Appliance on a Chip) [2] 에 대한 관심이 집중되고 있다. 이렇게 독자성을 가지고 발전되어온 구성 요소들이 하나의 시스템으로 고려되고 설계됨에 따라 설계의 복잡도, 설계 생산성 한계(Design Productivity Gap) [3] 등이 해결해야 할 중요한 문제로 대두되고 있다. 또한 시장이나 업계로부터의 다양한 요구에 신속하게 대응하기 위하여 디자인 재사용이나 하드웨어-소

프트웨어 통합설계 및 통합검증과 같은 다양한 개념이 연구 중이거나 이미 부분적으로 적용되고 있다. 이러한 급격한 시스템 설계 환경변화와는 달리 시스템 설계방법론은 아직 초보적인 단계에 머무르고 있는 것이 사실이다. 과거 10여 년동안 EDA 영역에서 다양한 시도가 이루어졌고, 아직 명확한 해법 또한 부재한 것이 사실이나, 활용 가능한 의미 있는 접근이 학계, 업계, 공식적인 단체를 통해서 꾸준히 이뤄지고 있으며 부분적으로 괄목할 만한 성과들이 선진사들을 통해 발표되고 있다. 따라서 기존의 설계 방법론의 한계를 인식하고 새로운 방법론에 대한 엄격한 고찰이 필요하다.

II. 과거의 설계방식

시스템 설계방식으로 폭포수모델(Waterfall Model)이 흔하게 적용되어 왔다. 그러나 SoC 같이 반복적이며 단계적인(iterative, incremental) 설계환경이 불가피한 경우, 객체지향에 근거한 디자인 재사용이 중요해지는 경우에 폭포수모델은 적당한 방법론이라고 할 수 없다 [4]. 더 나아가 기존의 H/W-S/W 설계방법은 시스템 전문가에 의해서 H/W-S/W가 조기 분할되고 독자적인 개발과정을 거친 후에 결합/검증 절차를 따르는 형태이므로 설계 후단에 발견되는 디자인 결함을 수정하는데 상당한 대가를 요하며 디자인의 복잡도의 급격한 증가와 더불어 기존 패러다임의 변화를 요구하게 되었다.

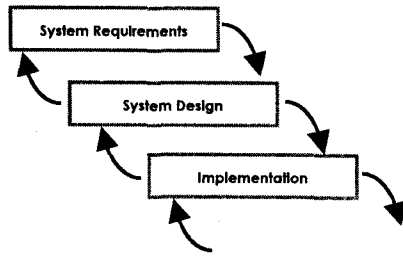


그림 1. 폭포수 모델

이러한 시스템 설계 패러다임 한계를 극복하기 위한 시도는 학계나 EDA 업체를 통하여 개별적으로 시도되기도 하였고 근래에 들어 여러 단체의 공식적인 노력을 하나로 묶는 표준화 과정이 (VSI Alliance) 진행되기도 한다. 먼저, 학계나 EDA 업체의 노력을 통합설계(co-design)와 통합검증(co-verification)의 분야로 나누어 살펴보면, 통합검증은 다음과 같이 세 가지 범주로 분류될 수 있다. [5]

1. ASIP(Application Specific Integrated Processor) Co-design
2. H/W-S/W Synchronous system co-design : *Vulcan, Codes, Tosca, Cosyma, etc*
3. H/W-S/W for distributed systems : *N2C, Siera, Specsyn, Ptolemy, Cosmos, VCC, CoMET, etc*

그리고 통합검증의 경우는 다음과 같다.

1. H/W simulator – S/W processor model 연결 커널형: *CVE, Eaglei, etc*
2. 상위 레벨 설계기능 확장형: *CoMET, N2C, etc*
3. H/W accelerator/emulator 의존형: *Xtreme, Quickturn, Celaro, etc*
4. H/W-S/W API, HDL interface library 제공형

이와 같은 개별적인 시도와 별도로 공식적인 표준화 노력으로 진행중이다. 이 중에 가장 대표적인 VSI Alliance(www.vsi.org)는 Hard/Firm/Soft Core 의 사용성을 높이기 위한 표준화와 새로운 설계 패러다임에 근거한 상위단계 설계표준화 등을 추진하고 있다. 이 때 상위단계 설계표준화에 SLIF(System Level Interface Behavioral

Documentation), OCB VCI(On-Chip Bus Virtual Component Interface), System-Level Data-Type Standard 가 해당된다 [6].

지속적으로 연구되어온 개별적 또는 공식적인 도구와 방법론의 이면에는 기존의 설계의 한계를 극복하기 위한 새로운 개념들이 존재한다. 첫 번째로 Interface-Based Design [7] 은 디자인을 순 내부 동작과 외부 인터페이스 동작으로 분리 설계하여 자유로운 파티셔닝을 진행하는데 도움을 준다. 다음으로 Platform-Based Design [8] 은 응용분야에 따라서 일반적으로 쓰일 수 있는 재사용 아키텍처(platform)을 설계에 이용하는 방법이다. 이 두 가지 개념은 설계의 구성요소를 분리하여 생각하는 직교화 설계 개념(orthogonal design)으로 설계의 효율을 높이기 위한 것이다. 설계의 관점 외의 검증의 관점에서 TBV(Transaction-Based Verification Methodology) [9], 가상 프로토타입(Virtual Prototype)의 개념들이 제시되었다. TBV 는 추상 단계가 높은 검증 코드를 사용하여 검증과정의 양적 / 질적 향상을 추구하는 방법이다. 과거에는 프로토타입을 보드에 구현하고 검증하였으나 최근은 가상 프로토타입을 이용한 검증을 통하여 짧은 시간에 손쉽게 검증과 디버깅이 가능하게 되었다.

지금까지 현재, 그리고 앞으로 닥치는 시스템 설계 경향과 문제점을 알아보았고, 과거에 이를 극복하기 위하여 연구되고 적용되었던 개별적 또는 공식적인 도구 및 방법론도 언급하였다. 또한 이러한 시도의 근간이 되는 근본 개념들에 대해서도 간략히 제시하였다. 이제는 이러한 추세에 따라 시스템 설계에 필요로 하는 몇 가지 원칙들을 언급하고, 당 랩에서 고려 중에 있는 방법론을 간략히 제시한다.

III. 새로운 설계방식

시스템 설계의 환경변화에 대응하기 위하여 새로운 방법론이 가져야 할 몇 가지 원칙을 아래와 같이 제시한다.

1. 객체지향에 준한 설계

시스템은 꾸준히 재사용되고 다양한 추상화 단계를 가져야 한다. 이를 가장 효과적으로 대응하기 위해서는 객체지향 설계방식이 유용하게 이용될 수 있다.

2. 다양한 모델을 수용하는 설계환경

앞으로의 시스템은 다양한 도메인을 포함된 형태가 될 것이다. 따라서 다양한 모델을 동시에 고려하고 수용할 수 있는 설계환경이 필수적이다.

3. 설계 과정의 불연속성 최소화

폭포수모델과 달리 반복적이고 단계적인 설계가 수행될 수 있도록 설계 단계의 불연속성을 최소화하는 설계방법론이 필요하다.

4. 정형 / 비정형 기술방법의 조화

설계과정에서 정형적인 기술과 비정형적인 기술은 공존이 불가피하다. 그러므로 정형과 비정형간의 상관관계를 명확히 정의하고 이해되도록 기술방법의 조화가 필요하다.

5. 풍부한 다면적 라이브러리 구축

하향식 설계방법에서나 디자인 재사용을 중심으로 한 설계에서도 다양한 추상단계 뿐만 아니라 여러 가지 아키텍처로 구현된 다면적인 라이브러리의 구축이 절실하다.

새로운 방법론을 위한 몇 가지 원칙에 준하여 지금까지 발표되어온 상용도구를 이용한 효과적인 시스템 설계 방법론을 간략히 제시한다. 그림 2에서 보는 바와 같이, 설계의 시작은 주어진 비정형 기술(Specification)로부터 시작하고 이 비정형 기술은 객체지향의 유사 정형기술로 변형된다. 이때 유사 정형 기술은 실행 가능하거나 자동변형이 가능하지는 않지만 이후에 전개되는 설계과정에서 비정형 기술과 정형기술(executable & formal specification)의 교량역할을 한다. 유사 정형기술에는 UML(Unified Modeling Language), 객체지향 설계방법론으로 Sintef 의 TIME [10] 등을 고려할 수 있다. 그러나 UML 의 역할에 대한 유효성은 아직 논의 중이다 [11]. 다음 단계로 구현하고자 하는 시스템의 기능동작(Behavior)와 아키텍처(Platform)를 앞서 설명한 직교개념에

따라 분리 설계한다. 이때 기능동작의 설계는 기술 언어나 특정 개발도구의(SPW, Cossap, SDL, VHDL 등) 구현 방법론에 따라 충실히 진행된다. 다만, Interface-Based Design, SLIF, OCB VCI, Platform-Based Design 등과 같은 개념의 원칙에 따라 설계한다. 그리고 기존의 설계된 IP 를 재사용하여 개발기간을 단축하도록 하여야 한다. 아키텍처의 설계는 설계하는 시스템의 응용분야에 따라 적절한 Platform 을 정하는 것으로 커디션이나 매핑을 통해 아키텍처의 효과를 빠르고 정확히 반영할 수 있도록 하는 모델링과 방법론이 필수적이다. 이때, 설계의 단계에 따라 정확성과 설계 효율성간의 적절한 손익관계를 고려해야 한다. 그리고 채널 아키텍처를 상세화 (refinement)할 때는 되도록 정해진 표준 라이브러리를 재사용하도록 한다. 그렇게 함으로써 다양한 디자인-영역/아키텍처 검토(design space / architecture exploration)를 빠르고 편리하게 진행할 수 있다. 지금까지의 설계과정을 마치면 가상 시스템(Virtual System)과 S/W 를 연계 되고 검증의 단계로 진행한다. 이때, 검증의 단계는 많은 시간과 노력을 필요로 하므로 H/W-S/W 모두 고성능의 검증모델을 사용하도록 환경을 구성하고, 때에 따라서 적절히 정확성을 양보하고 상위모델을 사용하는 것도 고려한다. 그리고 통합설계와 통합검증을 엄밀히 분리하여 설계하기 보다 통합설계 기능을 수용하는 환경(N2C, CoMET)을 이용하는 것이 설계의 불연속성을 제거하는데, 통합검증의 효율면에서 바람직할 것이다.

앞서 간략히 제시한 설계 방법론은 객체지향을 근간으로 하고 다양한 언어, 추상화 단계를 수용하는 설계환경으로, 각 설계단계가 엄밀히 분리된 도구나 방법론을 지양하고 서로 자연스럽게 교환 가능한 형태를 가지도록 함으로써 설계상의 불연속성을 제거하도록 하였다. 이때 발생하는 IP 는 재사용이 가능하도록 재설계의 개념을 도입하였다.

IV. 결론

기존의 시스템 설계 방법론은 최근의 시스템 경향에 대응하기에는 한계에 이르렀다. 그리고 이를 극복하기 위하여 학계, 업계, 또는 공식적인 단체를 통하여 제시된 여러 도구, 방법론, 개념을 살펴보았다. 이어서 앞으로 시스템 설계에 필요한 원칙들(객체지향설계, 이종모델수용, 불연속성 최소화, 정형/비정형 기술의 조화, 다면적 라이브러리 구축)을 제시하였다. 앞으로 현재의 상용 도구와 방법론으로 제시한 원칙을 만족하는 본문에서 간략히 보인 시스템 설계 방법론이 꼭 필요할 것으로 생각된다.

[1] ITRS, "International Technology Roadmap for Semiconductors, 1999 Edition, SYSTEM-ON-A-CHIP", Annual Report, 1999 (www.itrs.net)
 [2] <http://www.national.com> (Geode Products)
 [3] SRC, "Technology Requirements", SRC Presentation Material, International Conference on Microelectronic Systems Education, Arlington, Virginia, July 19, 1999
 [4] Bran Selic et al, *Real-Time Object-Oriented Modeling*, John Wiley & Sons. Inc., USA, 1994, pp

3-11

[5] Jorgen Straunstrup et al. "Hardware/Software Co-design: Principles and Practice" Kluwer Academic Publishers, 1997, pp. 309-310
 [6] Christopher K. Lennard et al. "Standards for System-Level Design: Practical Reality or Solution in Search of a Question ?" Design, Automation and Test in Europe Conference and Exhibition 2000. Proceedings, 2000
 [7] James A. Rowson et al. "Interface-Based Design" Design Automation Conference, 1997. Proceedings of the 34th
 [8] Martin G. McNelly, A., and Todd, L., "The integration platform approach to System-On-Chip Design," Proceeding of IP 98 Europe, Frankfurt, October 1998, pp. 101-116
 [9] <http://www.testbuilder.net/documentation/whitepapers/tbv00tr2.pdf>
 [10] <http://www.informatics.sintef.no/projects/time>
 [11] <http://www.eetimes.com/story/OEG20000929S0053>

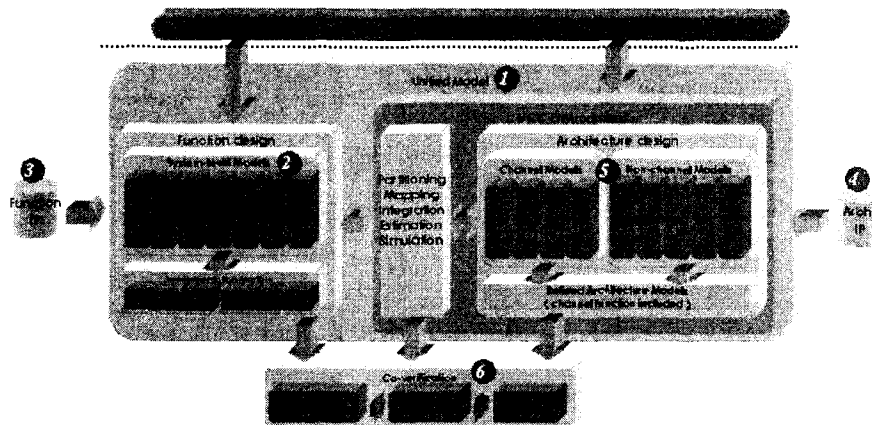


그림 2. 제안된 방법론