

기능 안전을 고려한 칩 설계 방법론 및 EDA Tool 동향

1. 서론

스마트, 컨넥티드 자동차의 급속한 발전으로 수백개의 IC와 여러 종류의 커뮤니케이션 네트워크에 1억 줄 이상의 소프트웨어 코드가 탑재되게 된다.^[1] 또한, nanometer 반도체 기술 발전으로 소자의 물리적인 한계가 심화되고, 한 칩에 수십억 개의 트랜지스터가 탑재될 수 있게 되어 복잡도가 크게 가중되었다. 점점 더 복잡해지는 시스템 및 칩 설계에 ISO 26262 표준 등의 기능 안전을 고려하는 것이 매우 중요한 이슈로 등장하였다.^[1-3]

하드웨어 및 소프트웨어의 설계 측면은 물론 검증 측면에서도 기능 안전을 고려해야 한다. 차량용 반도체의 기능 설계 및 검증에는 기존의 SoC (System-on-Chip) 설계 툴 및 방법론을 개선하여 사용하면 된다.^[4-6] 하드웨어 및 소프트웨어의 기능 안전을 고려한 설계 및 검증을 위하여 FMEDA (Failure Modes, Effects and Diagnostic Analysis) 및 FTA (Fault Tree Analysis) 등 다양한 기법들을 추가로 사용한다. Fault 모델링과 fault 주입 (injection) 기법을 통하여 오류 (error)를 파악하고, 기능 안전성의 검증을 수행한다.

스캔 테스트 및 BIST (Built-In Self-Test) 등의 DFT (Design for Testability) 기법과 다양한 DFS (Design for Safety) 기법을 활용하여 ISO 26262의 테스트 요구 조건을 만족시킨다. 성능과 기능 안전을 동시에 고려한 설계, 검증 및 테스트로 추가적인 노력과 시간이 투입되어야 하며, 이를 해소하기 위하여 효율적이고, 효과적인 설계, 검증 및 테스트 툴 및 방법론이 수립되어야 한다.

EDA (Electronic Design Automation) 3사인 시놉시스, 케이던스, 멘토를 포함한 회사들과 학계의 자동차용 칩 개발 툴 및 방법론 동향을 살펴보고자 한다. 수년전부터 DAC (Design Automation Conference)



공 정택
성균관대학교
소프트웨어대학 교수



등에서도 자동차업체의 관련 발표가 증가하고 있어, 이들도 간단히 소개한다.

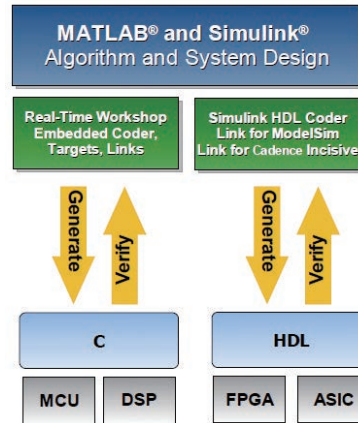
II. 기능 안전을 고려한 칩 설계 툴 및 방법론

1. 모델 기반 설계 (Model-Based Design)

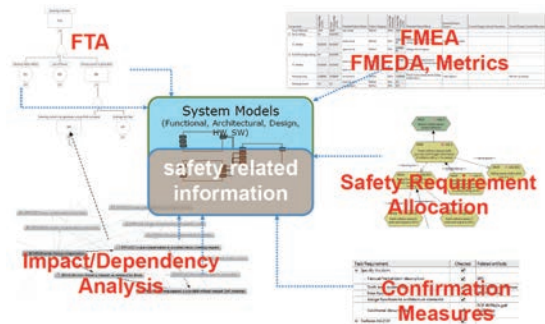
자동차 개발에 모델 기반 설계 방법을 필수적으로 사용하고 있다. 모델 기반 개발 방식은 고수준 언어 (high-level language)를 이용하여 시스템 레벨의 설계 및 검증을 진행하여 오류를 조기에 발견하고, 시스템의 하드웨어와 소프트웨어를 동시에 설계하고 검증하는 방법론이다. 모델 기반 설계에서는 시스템 아키텍처와 기능 동작 모델링에 MATLAB/Simulink와 dSPACE TargetLink 등을 주로 사용한다.^[6-8] Ford사는 모델 기반 설계에 UML (Unified Modeling Language)과 SysML (Systems Modeling Language)을 사용하며 재활용성이 매우 높은 모델 기반 플랫폼을 개발하여 활용하고 있다.^[9]

널리 활용되는 모델 기반 설계 방법 예를 <그림 1>에 나타내었다.^[10] TargetLink, Embedded Coder 등으로 C/C++ 코드를 자동 생성시키며, Cynthesizer, Catapult, HDL Coder 등의 고수준 합성 (high-level synthesis) 툴로부터 RTL (Register Transfer Level) 코드를 자동 생성시킬 수 있다. C/C++ 코드의 자동 생성이 수작업보다 훨씬 빠르며, 오류를 줄일 수 있다. 현재 자동차에 사용되는 소프트웨어의

30% 정도만 모델 기반 방식으로 개발되고 있으나, 73% 정도로 개발된다는 보고도 있다.^[2] 자동 생성된 코드는 수작업으로 개발한 것보다 성능이 떨어질 수 있으나, 개발 시간을 크게 단축할 수 있고, 오류를 줄일 수 있다. 재활용도와 개발 효율을 높이기 위하여 모델 기반의 소프트웨어 자동 생성이 계속 증가하는 추세이다. ISO 26262 표준의 ASIL (Automotive Safety Integrity Level) 수준 코드를 자동으로 생성하는 연구도 진행되고 있다. 자동 생성된 하드웨어 RTL 코드를 사용하여 FPGA를 합성



<그림 1> 모델 기반 설계 방법 예^[10]



<그림 2> 모델 기반 설계 방법을 활용한 ISO 26262 기능 안전 해석 예^[11]

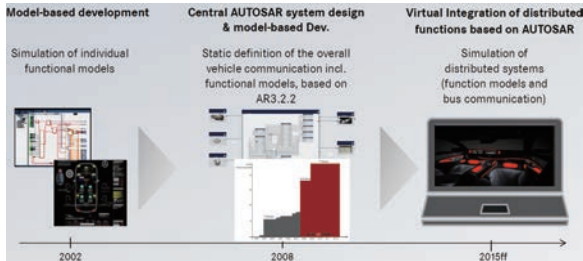
하여 프로토타이핑에 활용할 수도 있다.

Medini라는 해석 툴에 요구 사항 관리 및 모델 기반 설계 환경인 IBM Rational DOORS/Rhapsody와 MATLAB/Simulink 등이 연결되어, <그림 2>와 같이 기능 안전 해석에 활용되기도 한다.^[11-12] 기능 안전 및 신뢰성을 분석하여 HARA (Hazard Analysis and Risk Assessment), FMEDA, FTA 등과 연결되어 컴포넌트의 ASIL 결정에 활용된다.

2. 가상 프로토타입 (Virtual Prototype)

MATLAB/Simulink 등을 사용한 모델 기반 모델링에서 기능뿐만 아니라 타이밍, 전력 소모, 칩 면적 등의 성능도 모델링할 수 있다. ESL (Electronic System Level)에서 C/C++/SystemC 등을 활용하여 가상 프로토타이핑 (virtual prototyping)을 수행하여 하드웨어-소프트웨어

모델 기반 개발 방법이 시스템 레벨 설계 및 검증에 필수적으로 사용되고, C/C++ 소프트웨어 및 RTL 하드웨어 코드를 자동으로 생성하여 기능안전 해석에 활용된다.



〈그림 3〉 모델 기반 설계 방법이 AUTOSAR 표준 플랫폼기반으로 분산된 기능들을 통합하는 가상프로토타이핑 방법론으로 발전^[14]

어를 동시에 설계하고 해석할 수 있다. ESL 프로토타이핑은 TLM (Transaction-Level Modeling)을 기반으로 하며 다양한 추상화 레벨 (system specification, timed functional, bus cycle-accurate, computation cycle-accurate, cycle-accurate model 등)로 구현된다.

가상 프로토타이핑은 하드웨어-소프트웨어 동시 개발 및 검증, 설계 영역 탐색 (design space exploration) 등의 플랫폼으로 이용된다. 예로, Simulink 모델을 SystemC 모델로 자동 변환하고, 아키텍처와 매핑 작업을 통하여 가상 프로토타입을 생성할 수 있다. 이를 통하여 기능, 성능은 물론 안전성을 시뮬레이션으로 평가할 수 있다.^[13] 이를 vHIL (virtual Hardware In-the-Loop)이라고도 한다. vHIL 개념을 통하여 실제 칩 프로토타입이 개발되기 전에 가상 프로토타입 상에서 펌웨어 (firmware) 등을 개발할 수 있다.

시놉시스, 케이던스, 멘토 등 여러 회사에서 가상 프로토타이핑 툴들을 제공한다. 〈그림 3〉과 같이 기능 안전을 고려한 모델 기반 설계 방법이 자동차용 임베디드 소프트웨어의 재사용 표준 플랫폼인 AUTOSAR (AUTomotive Open System ARchitecture)를 기반으로 하는 통합 가상 프로토타이핑으로 발전하였다.^[14]

3. Fault 주입을 이용한 기능 안전 검증

기능 검증 (functional verification)은 시스템의 기능이 정상적으로 동작하는 지 검증하는 방법이나, 기능 안전 검증 (safety verification)은 fault가 생긴 시스템과 정상적인 시스템의 동작을 비교하는 방법이다. 기능 검

증 기법에 안전 검증을 포함시키는 방법이 소개되었다.^[15] Coverage 기반 기능 검증으로 무결점 설계를 수행하고, 기능 안전 검증을 통하여 안전 리스크를 없애야 한다. 초기 설계에서 코딩 가이드라인을 체크하고, 초기 검증에서 fault 모델링을 하는 등 다양한 DFS (Design for Safety) 방법을 활용해야 한다.

기능 안전 검증에 fault 주입 방법을 사용하며, 하드웨어 및 소프트웨어의 신뢰성 (dependability) 평가에도 활용한다.^[16] Fault 주입을 활용하면 소프트웨어의 테스트 커버리지가 높아진다. Fault 주입은 하드웨어 기반, 소프트웨어 기반, 시뮬레이션 기반, emulation 기반 기법 등으로 수행된다.^[16-17] 하드웨어 기반 fault 주입은 하드웨어 프로토타입이 반드시 존재해야 하고, 실제 오류 상황을 반영하는 제어성이 떨어지는 단점이 있다. 소프트웨어 기반 fault 주입은 하드웨어 fault 주입과 같은 효과를 제

ESL 프로토타이핑은 TLM을 기반으로 다양한 추상화 레벨로 구현되며, 기능, 성능 및 안전성을 시뮬레이션으로 평가할 수 있다.

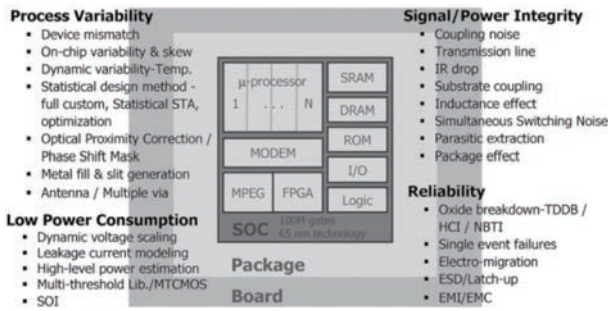
공할 수 있으나, 테스트성에 제한이 있고, 실제로 동작하는 소프트웨어와 다른 기능 및 성능을 보이는 문제점이 있다.

RTL 및 게이트-레벨 시뮬레이션 기반 fault 주입은 널리 사용되는 방법이나, RTL 설계 완료 후에 사용 가능하며, 시뮬레이션이 느리고, 복잡한 고장 (failure) 시나리오에 적용하기 어려운 단점이 있다. 시뮬레이션 속도를 높이기 위하여 FPGA emulation을 활용하기도 한다.

시뮬레이션 기반 fault 주입의 제약점을 극복하기 위하여, 상위 수준인 동작 모델 단계에서 fault 주입과 뮤테이션 (mutation) 테스트를 결합하여 시스템의 변화를 해석하는 모델 기반 테스트 방법이 소개되었다.^[18] TLM/emulation 혼용 환경을 사용하기도 한다. 가상 프로토타입 상에서는 fault의 영향 추적이 가능하고, 설계 초기 단계에서 안전성을 개선할 수 있는 장점이 있다. Fault 주입 지점, 종류, 주입 시점 및 종점, 고장 시나리오 등이 반영된 FMEDA 방법으로 활용 가능하다.

4. 신뢰성을 고려한 설계 (Design for Reliability)

차량용 반도체는 가전 제품보다 더 높은 온도에



〈그림 4〉 Nanoscale 반도체 칩의 신뢰성 확보를 위하여 해결해야 할 문제들^[4]

서 15년 이상의 수명과 무결함 (zero defect)이 보장되어야 한다.^[19] 반도체에는 다양한 종류의 결함이 생길 수 있으므로 이를 극복할 수 있도록 칩이 개발되어야 한다. Nanometer 반도체의 공정 변동에 기인한 RDF (Random Dopant Fluctuation), LER (Line Edge Roughness), 게이트 절연막 두께 변동 등이 고려되어야 한다. 사용 시간이 경과됨에 따라 발생하는 HCI (Hot-Carrier Injection), TDDB (Time-Dependent Dielectric Breakdown), NBTI (Negative Bias Temperature Instability), EM (Electromigration) 등도 고려되어야 한다.^[20-21] 신호/전력 보존성 (signal/power integrity) 강화를 위하여 동적 전압 강하 (dynamic voltage drop)를 고려한 타이밍 및 노이즈 분석이 필요하다. 전력 공급선의 EM 해석, RDI (Redundant Via Insertion), 안테나 효과 용다 이오드 삽입 방법 등도 사용된다.^[19] 차량용 반도체에서는 ESD (Electrostatic Discharge), EMI (Electromagnetic Interference)와 RISE (Radiation-Induced Soft Error)를 고려한 설계도 중요하다.

〈그림 4〉와 같이 기능 안전이 확보된 nanoscale 반도체 칩 설계를 위하여 공정 변동에 기인하는 신뢰성은 물론, 신호/전력 보존성, 저전력 소모 등의 문제를 트랜지스터 단계부터 시스템 소프트웨어까지 모두 해결하는 혁신적인 CAD (Computer-Aided Design) 툴 및 기법이 필요하다.^[4, 21] 반도체 칩 개발 시 신뢰성 및 품질 향상을 고려하는 것에 기능 안전성을 추가로 고려해야 한다. 그

동안 크게 중요시되지 않았던 저전력 설계 방법도 전기 자동차 시대로 진입하면서 더욱 중요해진다.

5. 기능 안전이 고려된 아키텍처

기능 안전 강화를 위하여 redundancy와 체커를 기본으로 하는 다양한 fault 감내 (tolerant) 기능을 탑재한다. ECC (Error-Correction Code), 패리티 (parity) 검사, CRC (Cycle Redundancy Check), DCLS (Dual-Core Lock-Step), 클럭 및 전압 모니터링, 타이머 등이 대표적인 예이다. 메모리 및 로직 BIST, 스캔 테스트, ATPG (Automatic Test Pattern Generation) 등의 DFT 기법도 적용한다. 또한, CCF (Common Cause Failure)를 최소화할 수 있도록 논리적, 물리적인 설계 최적화를 수행한다. 반전된 신호, 2-클럭 지연, 전력선 및 신호선 분리, 크로스톡 제거 방법들이 활용된다.

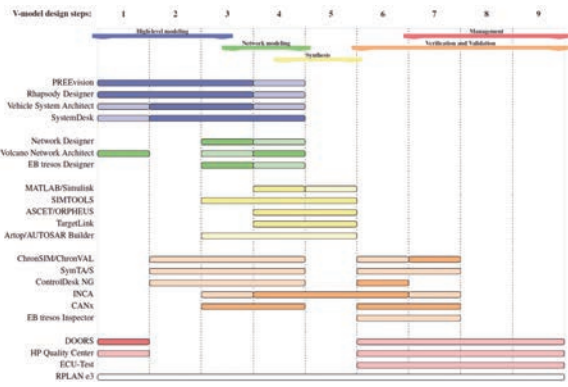
최근, 인텔로 인수된 Yogitech에서는 fRMethodology (faultRobust Methodology)라는 기능 안전 방법론을 제공한다.^[22] 오류에 민감한 회로 (sensible zone) 순위에 따라서 redundancy 설계

정도를 조정한다. 예로, DCLS에 사용되는 두 개의 코어 대신에 fRCPU 코어를 사용한 OTC (Optimized Tightly Coupled) dual 코어를 잘 사용하면, 칩 사용 면적을 크게 줄일 수 있다. 기능 안전을 강화하기 위하여 검증된 IP와 소프트웨어의 재사용도 권장한다.

6. SEoC (Safety Element out of Context) 플랫폼

경쟁력을 확보하기 위하여, 자동차 제조사의 요청 전에 반도체를 미리 개발해야 하는 경우가 발생한다. 자동차 제조사에서 최종 시스템에서 요구되는 기능 안전 사항이 정해지지 않은 상황에서는 반도체 공급사는 적합한 가정을 통하여 칩을 개발하게 된다. 이 가정된 시나리오의 적절성 및 정확성이 매우 중요하다. 개발된 후에 가정과 실제가 일치하는 지 확인하는 절차가 필요하다. ISO 26262에서 중요하게 사용되는 SEoC 플랫폼을 지원하기 위해서는 계층적 하향 (top-down) 및 상향 (bottom-up) 개

Fault 주입으로 기능안전 검증을 수행한다. 신뢰성이 고려된 설계, 기능안전이 고려된 아키텍처, DFT 및 기능안전 방법론이 필요하다.



〈그림 5〉 5개의 설계 단계와 ISO 26262에서 정의된 9개의 V-모델 개발 단계에서 사용되는 툴들의 기능과 성능 비교^[26]

발 방법론을 모두 준비해야 한다.^[11-12, 23-24]

7. 하드웨어와 소프트웨어 설계 및 검증 툴

EDA 3사 외에 Vector, LDRA, ETAS, dSPACE, IBM, ANSYS, KPIT 등 다양한 ECU 설계 및 검증 툴들이 소개되었다.^[11, 24-26] 〈그림 5〉

와 같이 5개의 설계 단계와 ISO 26262에서 정의된 9개의 V-모델 개발 단계에서 사용되는 22가지 상용 툴들의 기능과 성능을 평가하였다.^[26] 고품질 소프트웨어 개발을

위하여 요구 사항 분석, 모델 기반 설계 및 분석공학, 정적/동적 코드 분석, fault 주입 및 테스트 자동화, 취약성(vulnerability) 분석 등 소프트웨어 라이프 사이클 관리 툴을 사용한다.

8. 오픈소스 소프트웨어 플랫폼

소프트웨어 개발 효율을 높이기 위하여 오픈소스 소프트웨어를 사용하는 것이 대세이며, 임베디드 시스템 개발에 널리 활용되고 있으나, ISO 26262의 기능 안전 요구 조건들을 만족시키기 어렵다.^[2] BMW사에서는 Linux 등의 오픈소스 소프트웨어를 기반으로 하고 AUTOSAR 등과 연계된 차량용 오픈소스 소프트웨어 플랫폼을 개발하였다.^[27] 안전 모델링 및 분석 오픈소스 툴 플랫폼인 SAFE (Safe Automotive soFtware architEcture) 프로젝트도 진행되었다.^[28] 자동차 멀티-코어 시스템용 모델

기반 오픈소스 개발 환경도 소개되었다.^[29]

III. 결론

점점 더 복잡해지는 시스템 및 칩 설계에 ISO 26262 표준의 기능 안전을 고려하는 것이 매우 중요한 이슈로 등장하였다. 특히, 시스템의 복잡도 증대로 기능 안전 기술이 반도체 및 IP로 집중되는 추세이다. 이를 해결하기 위한 EDA 3사 및 기타 회사들과 학계의 차량용 반도체 설계 툴 및 방법론을 소개하였고, 주요 차량용 반도체 회사들의 기능 안전 대응 방안도 소개하였다. 차량용 반도체 하드웨어/소프트웨어 개발에는 기존의 반도체 설계 툴 및 방법론을 개선하여 적용하면 된다.

모델 기반 설계 및 가상 프로토타이핑을 통한 하드웨어/소프트웨어 개발 및 검증 방법을 소개하였다. 다양한 추상화 단계에서 수행되는 fault 주입을 이용한 기능 안전 검증을 소개하였다.

다양한 ECU 설계 및 검증 툴들이 있으며, 기능안전을 위하여 DFSS (Design for Safety and Security) 설계 툴 및 방법론을 준비해야 한다.

반도체 칩의 신뢰성 및 품질 향상을 위한 방법론 강화가 필요하며, 기능 안전을 추가로 고려하는 혁신적 방법론을 확보해야 한다. SEooC 플랫폼 및 다양한 설계 및 검증 툴

들의 평가도 소개하였다. 개발 효율 향상을 위한 오픈소스 소프트웨어를 ISO 26262 기능 안전에 적용하는 노력들을 소개하였다.

기존의 반도체 설계 툴 및 방법론이 메모리, 시스템 반도체 성공에 필수적이듯, 차량용 반도체의 기능 안전을 위하여 DFSS (Design for Safety and Security) 설계 툴 및 방법론도 확고히 준비해야 한다.

감사의 글

본고는 미래창조과학부가 지원하는 소프트웨어 중심 대학 지원사업과 미래창조과학부 산하 정보통신기술진흥센터가 지원하는 IT·자동차융합 ITRC 사업의 지원으로 수행되었습니다.



참고 문헌

- [1] M. Santarini, "Synopsys Automotive: From Silicon to Software to Smarter, Safer, and More Secure Driving," Synopsys Insight, Issue 1, pp.2–13, 2016.
- [2] A. M. Meroth, F. Trankle, B. F. Richter, M. Wagner, M. Neher, and J. Luling, "Functional Safety and Development Process Capability for Intelligent Transportation Systems," IEEE Intelligent Transportation Systems Magazine, pp. 12–23, Winter 2015.
- [3] 위재경, 채승엽, "기능 안전을 준수한 차량용 반도체 개발을 위한 기술적 고려 사항," 전자공학회지, 제 41권 제 1호, pp. 60–68, 2014년 1월.
- [4] J.-T. Kong, "CAD for Nanometer Silicon Design Challenges and Success," IEEE Tr. on Very Large Scale Integration (VLSI) Systems, vol. 12, pp. 1132–1147, Nov. 2004.
- [5] P. Boulet, "System-on Chip Design, Embedded System Design Challenges," 2009, <http://www.slideshare.net/pboulet/socdesign>
- [6] N. Ajward, "Evaluation of Automatic Code Generation Tools," Lund Univ., Apr. 2007.
- [7] M. Di Natale and A. L. Sangiovanni-Vincentelli, "Moving From Federated to Integrated Architectures in Automotive: The Roles of Standards, Methods and Tools," Proc. IEEE, vol. 98, No. 4, pp. 603–619, Apr. 2010.
- [8] T. Dalon, "AVL Embedded Software Model-Based Design Platform Based on MATLAB and Simulink," MathWorks Automotive Conf., Sept. 2015.
- [9] K. Post, G. Walley, and J. Che, "Integrating Descriptive Models with an Analytical Model Culture – Lessons Learned at Ford," INCOSE IW 2014/ MBSE Workshop, Jan. 2014.
- [10] P. J. Mosterman, "MATLAB and Simulink for Embedded System Design," 2007, <http://msdl.cs.mcgill.ca/people/mosterman/presentations/date07/tutorial.pdf>
- [11] O. Kath, "Trends and Challenges in Functional Safety – A European Perspective," May 2014, http://mdstec.com/main/seminar/2014AutoSW/dw/2014AutoSW_T1_5.pdf
- [12] D. Makartetskiy, D. Pozza, and R. Sisto, "An Overview of Software-based Support Tools for ISO 26262," Int. Workshop on Innovation Information Technologies – Theory and Practice, pp. 132–137, 2010.
- [13] M. Streubuhr, M. Jantsch, C. Haubelt, and J. Teich, "From Model-based Design to Virtual Prototypes for Automotive Applications," Embedded World Conf., pp. 1–10, Mar. 2009.
- [14] T. Ringler, "Virtual Integration and Test of AUTOSAR System at Daimler – Body & Comport Domain," Vector Congress, 2014.
- [15] A. Sherer, J. Rose, and R. Oddone, "Ensuring Functional Safety Compliance for ISO 26262," Design Automation Conf. (DAC), pp. 1–3, 2015.
- [16] M. Kooli and G. Di Natale, "A Survey on Simulation-Based Fault Injection Tools for Complex Systems," Int. Conf. on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), pp. 1–6, 2014.
- [17] S. Lee and S. Shin, "Software Fault Injection Test Methodology for the Software Verification of ISO 26262 Standards-based," Tr. of KSAE, vol. 22, No. 3, pp. 68–74, 2014.
- [18] R. Rana, M. Staron, C. Berger, J. Hansson, M. Nilsson, and F. Torner, "Improving Fault Injection in Automotive Model Based Development using Fault Bypass Modelling," Workshop on Software-Based Methods for Robust Embedded Systems, Sept. 2013.
- [19] A. Kashyap, S. Grimpen, and S. Sundaramoorthy, "Achieving Power and Reliability Sign-off for Automotive Semiconductor Designs," Design Automation Conf. (DAC), pp.1–6, 2015.
- [20] E. Maricau and G. Gielen, "Analog IC Reliability in Nanometer CMOS," Springer, pp. 11–35, 2013.
- [21] G. Georgakos, U. Schlichtmann, R. Schneider, and S. Chakraborty, "Reliability Challenges for Electric Vehicles: From Devices to Architecture and Systems Software," Design Automation Conf. (DAC), pp.1–9, 2013.
- [22] R. Mariani, G. Boschi, and F. Colucci, "Using an Innovative SoC-level FMEA Methodology to Design in Compliance with IEC61508," Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 492–497, 2007.
- [23] A. Ruiz, A. Milzi, and T. Kelly, "Systematic Application of ISO



26262 on a SEooC: Support by Applying a Systematic Reuse Approach,” Design, Automation & Test in Europe Conference & Exhibition (DATE), pp. 393–396, 2015.

[24] K. D. Muller–Glaser, “Model–Driven Engineering for Automotive Systems,” UCSD SAASE, 2009.

[25] P. Waszecki, M. Lakasiewicz, A. Masrur, and S. Chakraborty, “How to Engineer Tool–chains for Automotive E/E Architectures?,” ACM SIGBED Review 10(4):6–15, Dec. 2013.

[26] R. Johansson, J. Borg, J.–T. Kong, and B. C. Kim, “Enabling Safe and Innovative Electrical/Electronic (E/E) Systems in the Automotive Domain,” Auto Journal, Korean Society of Automotive Engineers, pp.45–51, Oct. 2014.

[27] L. Bulwahn, T. Ochs, and D. Wagner, “Research on an Open–Source Software Platform for Autonomous Driving Systems,” Oct. 2013, <http://www.bmw-carit.com>

[28] The SAFE Technology Platform – An Open Source Tool Platform for Safety Modeling and Analysis, 2013, <https://itea3.org/>

[29] P. Dziurzanski, A. K. Singh, L. S. Indruiasiak, and B. Saballus, “Benchmarking, System Design and Case–studies for Multi–core based Embedded Automotive Systems”, Int. Workshop on Dynamic Resource Allocation and Management in Embedded, High Performance and Cloud Computing DREAMCloud, Jan. 2016,



공정택

- 1981년 2월 한양대학교 전자공학과 학사 졸업
- 1983년 2월 연세대학교 전자공학과 석사 졸업
- 1994년 12월 Duke University Electrical Engineering 박사 졸업
- 1983년 1월~2006년 12월 삼성전자 CAE팀장, 상무
- 2006년 12월~2010년 12월 삼성전자 지적재산팀장, 전무
- 2010년 12월~2013년 12월 삼성전자 공과대학교, 부총장, 전무
- 2014년 3월~2015년 12월 한양대학교 융합전자공학부 교수
- 2016년 1월~현재 성균관대학교 소프트웨어대학 교수

〈관심분야〉

CAE, SoC, 차량용 반도체, IoT/임베디드 시스템