

# 임베디드 소프트웨어 유지보수 노력의 영향요인 연구 : 반도체 웨이퍼 가공라인 사례를 중심으로

조남형\*, 김치린\*\*, 김미량\*\*\*

성균관대학교 경영연구소\*, 어플라이드 머티어리얼즈 코리아\*\*, 성균관대학교 컴퓨터교육과\*\*\*

## Factors Influencing the Efforts for Embedded Software Maintenance : A Case from Semiconductor Wafer Processing Line

Namhyung Cho\*, Chi Rin Kim\*\*, Mi Ryang Kim\*\*\*

Institute of Management, SungKyunKwan University\*

Applied Materials Korea\*\*

Department of Computer Education, SungKyunKwan University\*\*\*

요 약 반도체 산업은 임베디드 소프트웨어를 통해 운영·통제되는 자동화설비를 통해 첨단상품을 생산한다. 반도체를 생산하는 로봇과 각종 설비의 임베디드 소프트웨어 유지보수는 제품의 품질과 신뢰성 제고를 위한 필수적인 과정으로 반도체 장비의 라이프 사이클을 고려할 때 상당히 높은 비중을 차지하는 활동영역이다. 그러나 이 분야에 대한 학술적 관심사는 그리 높지 않는데, 본 연구에서는 반도체 웨이퍼 생산장비를 구동하는 소프트웨어 관련 문제로 보고된 사건을 대상으로 502개의 데이터를 무작위 추출방식으로 수집하여 임베디드 소프트웨어의 유지보수 노력에 영향을 미치는 요인들을 분석해 보았다. 결론으로 실무적인 시사점도 제시하였다.

주제어 : 반도체, 임베디드 소프트웨어, 회귀분석, 소프트웨어 유지보수, 웨이퍼 가공

**Abstract** The semiconductor industry develops and maintains software embedded in computer-controlled tools and facilities, to process and manufacture high-tech products. Upgrading embedded softwares for semiconductor processing robots and machinery is one of the basic activities that must be performed in order to maintain product quality and integrity. Maintenance and enhancement of embedded software consume a major portion of the total life cycle cost of a system. However, the area has been given little attention in the literature. 502 maintenance and enhancement cases, related to embedded softwares in wafer processing machines, were selected at random for analysis. Practical implications are also discussed.

**Key Words** : Semiconductor, Embedded software, Regression, Software maintenance, Wafer processing

Received 21 July 2017, Revised 30 August 2017

Accepted 20 September 2017, Published 28 September 2017

Corresponding Author: Mi Ryang Kim

(SungKyunKwan

University)

Email: mrkim@skku.ac.kr

ISSN: 1738-1916

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. 서론

반도체 산업은 우리나라를 대표하는 기간 및 전략사업으로 2016년 기준 전체 무역흑자 898.3억달러 중 28.5%가 반도체 부문에서 달성된 것이며, 2017년 6월 현재, 메모리 반도체에서만 271.7억달러를 달성 중에 있을 정도로 국가경제에서 차지하는 비중이 매우 높다. 반도체 산업 내에는 칩 제조를 위한 다양한 장비산업이 포함되며 대규모 자본투자가 상시적으로 요구되는 장치산업이다. 반도체칩은 가로 세로 1cm 내외의 얇은 실리콘 웨이퍼 위에 트랜지스터, 캐패시터 등 각종 소자를 집적하여 만들며, 산술연산, 정보기억, 제어 등의 다양한 기능을 수행한다. 반도체 칩생산에 투입되는 장비는 수억에서 수백억을 호가하며, 수많은 디바이스와 로봇, 센서 등이 정밀하게 연계되어 소프트웨어 옵션을 바탕으로 유기적으로 운영된다. 장비의 운용은 크게 하드웨어와 소프트웨어 그리고 공정 분야로 구분될 수 있다. 하드웨어 분야는 하드웨어의 조립과 유지보수를 중심으로 이루어지며, 소프트웨어는 장비의 기능 및 자동화 운용 측면에서 필수적인 도구이며, 공정은 반도체 장비의 생산성 제고 프로세스에 주된 관심을 갖는다. 특히 반도체 장비운영에 필수적인 소프트웨어(Software, S/W)는 임베디드적인 성격을 주로 띠고 있기 때문에 장비 운용상의 문제점 해결이나 프로세스 개선은 S/W 개발 및 유지보수를 필요로 한다.

인공지능(AI), 사물인터넷(IoT), 클라우드 컴퓨팅(Cloud computing) 등 지능정보기술을 기존 산업이나 신기술에 접목시키는 4차 산업혁명이 현실화되면서, 반도체 장비 운용 역시 IoT 기반의 무인자동화쪽으로 급속히 발전하고 있다. 장비와 자재의 자동제어가 일반화 되어 가면서 다수의 장비를 동시에 운용·제어할 수 있는 S/W도 개발되고 있다. 이와 같이 장비구동에 필요한 S/W에 새롭고 복잡한 기능이 추가될수록 S/W의 복잡도 역시 가속화되고 있다. 장비의 운영목적에 따라 하나의 S/W에 여러 가지 옵션이 추가되면서 공정의 생산성 향상을 위해 순차적으로 추가된 옵션과 설정에 대한 최적화가 필수불가결한 요소가 되고 있다. 그 만큼 반도체 장비운용을 위한 임베디드 S/W의 중요성이 확대되고 있다. 반도체 장비의 기술수준이 급속도로 첨단화되면서 임베디드 S/W 시스템에 탑재된 기능과 옵션 또한 복잡하고 다

양화되고 있어 S/W 기반의 제어기술은 최종제품의 성능과 품질 및 생산단가에 지대한 영향을 끼치게 된다.

스마트폰이나 PC의 성능이나 품질이 운영체제에 의해 상당 부분 결정되었듯이[1, 2], 반도체 장비 역시 임베디드 S/W 시스템의 기능에 의해 생산성과 품질이 좌우된다. 생산품목이 복잡다양해지고 기술집적도가 심화되면서 이를 생산하는 장비에 수반되는 S/W의 보수·유지는 새로운 장비 개발시 요구되는 S/W 개발만큼이나 많은 시간과 인력투입을 요구한다[3].

따라서 반도체 장비 임베디드 S/W의 품질향상 측면에서 S/W의 수정과 변경과정에 수많은 요청사항을 처리 하였던 경험이 기록된 데이터베이스를 활용하는 통계적인 접근이 필요하다. S/W 문제가 최초로 제기되고 해결되는 과정 중에 발생하고 추가되는 제반 특성요인에 대한 객관적 검토기회가 될 수 있으며, 관리 측면에서도 축적된 데이터베이스 활용도를 높여 궁극적으로 업무처리의 능률 향상으로 이어질 수 있기 때문이다.

선행연구에서는 반도체용 임베디드 S/W 개발인력 육성과 수급에 관련한 논의[4, 5]와 국내 임베디드 S/W 산업분야전반에 걸친 기술력과 역량제고에 관련한 거시적인 측면에서 논의[6, 7]가 주로 있었다. 그러나 본 연구는 반도체 생산장비 임베디드 S/W 개발/유지보수과정을 단 위기업의 운영관리 측면에서 논의해보고자 하였다. 본 논문은 장비 S/W 기반으로 발생하는 각종 이슈 정보가 담긴 데이터베이스에서 제공되는 사안해결 기간과 이와 관련이 있을 것으로 기대되는 여러 가지 요인들을 선정하여 실증분석연구를 진행하였다.

연구자료 추출에 사용된 데이터베이스는 2012년도부터 2016년까지 5년간 300mm 웨이퍼 반도체 생산 전체 공정의 장비 임베디드 S/W에 관련하여 발생한 각종 사안의 해결과정 중 수집 및 기록된 운영관리적 측면의 축적자료이다. 실증분석의 결과는 기업 내 조직자원배분이나 운영관리 상 의사결정에 도움을 주는 객관적 지표로 활용될 수 있을 것으로 기대되며, 각 요인의 영향력을 가늠하여 S/W 문제사항이나 추후 반도체 장비시스템의 효율적 운용을 위한 의사결정지원도구로도 사용될 것으로 전망한다.

## 2. 반도체 칩 제조공정 및 임베디드 소프트웨어 운영환경 개요

### 2.1 반도체 제조공정 개요

반도체 칩은 실리콘으로 만들어진 둥근 원판 모양의 웨이퍼를 원자재로 사용한다. 삼성과 같은 최고의 반도체 제조업체들은 300mm 웨이퍼를 가공하는 설비를 주력으로 하고 있으며, 200mm 생산 장비는 반도체 칩 메이커 업체들을 위한 유지보수 측면에서 그 명맥을 이어가고 있다. 반도체 칩 제조공정은 회로설계 단계를 거쳐, 웨이퍼를 가공하여 하나의 칩으로 만드는 전 공정과 만들어진 칩을 보호하기 위한 패키징 단계인 후공정 단계로 구성된다.

본 연구는 웨이퍼를 생산 및 가공하는 전 공정 설비군에 초점을 두는데, 전체 공정은 식각(etching), 금속화(metalization), 열처리(Rapid Thermal Processing), 에피택시(epitaxy), 증착(deposition), CMP (chemical mechanical polishing) 등의 여섯 공정으로 구성된다. 식각 공정은 웨이퍼에서 특정 물질을 제거하는 공정이며 웨이퍼 전면부의 물질로 덮여있는 마스크(mask)의 모양새에 따라 특정 패턴이 정의된다. 마스크 물질로 보호되지 않는 웨이퍼와 재료는 액체 화학물질을 이용하여 식각하거나, 플라즈마 에칭가스를 이용하여 작업하기도 한다. 금속화공정은 웨이퍼에 증착되기 위한 재료소스로부터 증발된 물질이 진공 또는 저압 가스 환경을 통해 증기 형태로 응축되어 막(film)형태로 웨이퍼 표면에 증착되는 과정이다. 금속 열처리 공정을 거친 후에는 웨이퍼 기관의 전면부인 상부에 원자 단위의 결정학적 구조를 가지는 결정을 규칙적인 배열로 진행하여 해당 단결정을 성장시키는 공정에 진입한다. 다섯 번째 공정은 증착공정으로 증기의 화학 반응으로부터 가열된 웨이퍼 표면에 고체를 증착시키는 과정이며, 마지막 CMP공정은 산화규소, 금속 및 폴리 실리콘 등을 웨이퍼 표면에서 제거하여 웨이퍼 표면을 부드럽고 평탄하게 만드는 과정이다 [8].

반도체 제조 장비는 여러 개의 모듈로 구성되어 있는데, 모듈을 세분화하면 우선 웨이퍼에 가공이 이루어지는 과정인 프로세싱 챔버(processing chamber), 웨이퍼가 투입되고 반출되는 EFEM(equipment front end module), 웨이퍼 투입 후 프로세싱 챔버까지 장비내에서

의 이동을 담당하는 모듈 TM(transfer module), 일시적인 보관장소인 버퍼(buffer), 전력관련 시설 및 가압 펌프, 공정별 필요 물질 공급시스템 등으로 구분할 수 있다.

### 2.2 장비운용을 위한 임베디드 소프트웨어

반도체 장비에 내장된 임베디드 S/W는 칩 제조의 핵심적인 부분이다. 과거에는 펌웨어(firmware)나 보편적인 장비 운용에 필요한 도구 정도로 인식하였지만 고객요청에 의한 신규 플랫폼 출시 시간이 단축되면서 임베디드 S/W의 중요성은 매우 커지고 있다.

반도체 제조공정에 사용되는 장비에는 개별적으로 컴퓨팅 시스템이 탑재되어 있으며 운영체제를 기반으로 별도의 장비운용 어플리케이션 S/W가 구동되며, 하드웨어와 S/W간의 인터페이스를 담당한다. 운영체제는 장비제조사 별로 상이하지만 윈도우 기반, 리눅스 또는 유닉스 기반의 운영체제가 탑재되어 있다. 운영체제를 구동시키는 컴퓨팅 환경은 단일 서버를 통해 각 어플리케이션을 중앙처리하거나 복수의 서버급 하드웨어를 이용하여 분산처리 작업이 이루어진다.

반도체 장비용 임베디드 어플리케이션은 매우 다양한데, 자동화 통신 담당, 데이터를 기록하기 위한 데이터베이스, 로드포트 처리, 로봇구동, TM 운용, 프로세싱챔버 운용, GUI(graphical user interface), 하드웨어 구동을 담당하는 RT S/W(real time software), 백신, 다수의 어플리케이션 연계, 모니터링 툴 등을 예로 들 수 있다. 특히나 생산성을 높이기 위해 장비의 클러스터화가 진행되면서 S/W를 통한 운영관련 이슈의 처리 난이도도 증가하게 되었다. 웨이퍼의 장비간 무인 운송이 원격제어를 통해 가능해졌다는 점도 임베디드 S/W의 적절한 보수·유지가 더욱 중요한 문제로 대두되고 있다.

### 2.3 임베디드 소프트웨어의 보수·유지 니즈

반도체 칩 제조공정의 가동이 중단되거나 품질 이상, 또는 특정 기능이 오동작하는 경우 문제의 원인은 하드웨어, S/W, 공정 등의 세 가지 측면에서 추적하게 된다. 통상적으로 하드웨어 또는 S/W 문제로 간주할 수도 있지만 제조공정 특성에 부합하는 복합적인 요건을 준수해야 하는 문제가 나타나는 경우 이를 공정측면의 문제인 식으로 볼 수 있다.

반도체 장비에서 발생하는 S/W에 관련한 사안들은

정확한 원인이 밝혀지기 전까지는 임베디드 S/W 특성상 해당 원인이 무엇으로부터 발생하였는지 직관적으로 판단하기 어렵다. 반도체 장비 S/W 에 제기되는 문제는 다양한 형태로 발생된다. 동일 이슈가 반복되는 경우가 있는 반면에 단일하게 발생하는 경우도 종종 발생한다. 사안의 발생원인이 명확하지 않아 S/W, 하드웨어, 공정 중 둘 이상의 부문에 걸쳐서 나타나는 사례도 존재한다. GUI에서 표시되는 글자 일부의 수정 및 추가와 같은 단순한 사례도 있지만, 장기간에 걸친 특정 어플리케이션의 메모리 할당 문제로 인한 리소스 점유율(resource occupancy) 증가 등과 같은 문제는 수정을 위한 구조적인 검토 및 재디자인이 필요한 경우이다.

S/W 관련 문제를 해결하지 못하면, 웨이퍼 프로세싱을 사용자가 원하는 품질 수준에 맞추어 편차 없이 진행하기 어렵다. 웨이퍼가 투입되어 가공, 반출되는 과정에 사용되는 정보에는 반도체 칩생산에 필요한 가공공정에 대한 정보를 담고 있는 Recipe정보, 이를 어떤 시점에 어느 모듈에 위치하고 어디로 이동 혹은 대기시킬지 등 진행에 대한 Sequence정보, 그리고 각종 장비환경에 최적화된 설정값 정보 등이 포함된다. S/W는 이러한 정보를 활용하여 장비를 작동시키는데 이 과정에서 다양한 이슈 혹은 문제점이 제기된다.

예를 들어 스케줄러에 관련된 사안은 S/W 자체 문제인 경우가 대부분이다. 스케줄러는 주변 모듈의 동작시간 등 공정조건을 고려하여 웨이퍼가 별도의 시간 낭비 없이 최적화된 경로로 이동하도록 수립된 알고리즘이다. 웨이퍼 가공은 이 알고리즘을 바탕으로 진행되므로, 웨이퍼가 진행 중 하드웨어 이상 없이 멈추는 경우나 대상 Recipe가 적용될 시점이 아닌 경우, 혹은 장비운영회사의 처리율(throughput) 개선 요청 등이 이슈화 될 수 있는 사안이다.

S/W 관련 문제로 제기된 사안이었지만 실제적으로 하드웨어 관련 이슈인 것으로 판단된 사례로는 로봇의 상태와 구동 문제에 관한 것을 들 수 있다. 예를 들어 로봇 제어 시 로봇이 정상적인 위치에 위치하지 않거나 로봇 상태에 대한 정보가 비정상적인 것으로 나타날 때, 이 정보가 로봇 자체에 내장된 S/W에서 발생하는 경우는 하드웨어 문제로 판단할 수 있다. 하지만 작업자에게 공정정보로 제공되는 자료나 정보의 구조나 패턴에 오류가 있는 경우에는 S/W 문제로 볼 수 있을 것이다.

공정 측면에서 제기되는 S/W에 관한 사안은 일반적으로 반도체 프로세싱 챔버(chamber) 내의 웨이퍼를 대상으로 공정 처리 진행 중에 나타나는 이슈 혹은 광범위한 공정 처리 측면의 문제에 관한 것이다. 예를 들어 웨이퍼 공정 진행 후 온도의 불균일성이 품질 문제로 대두되었을 때, 챔버 히터로부터 수반되는 온도수치가 인가되는 시점에 일부 시간차가 있는 것으로 본다. 이는 S/W가 내리는 명령 처리시점이 문제의 원인일 수 있는 부분이다.

위와 같은 사례들은 공정별 또는 장비별로 다양한 조합으로 발생할 수 있다. 임베디드 S/W 내 포함되는 설정값 및 센서 값들은 종래에 들어서 그 숫자만 수만 가지 이상이며 지금도 지속적으로 개발 및 추가되고 있는 실정이다. 어플리케이션, 모듈, 공정 진행 특성 별 설정값을 조합하면 S/W가 보여줄 수 있는 결과 값 및 동작방식의 경우의 수는 매우 커질 것이며 이는 경험적으로 사안의 해결 난이도를 상승시키게 된다.

### 3. 소프트웨어 이슈 및 보수·유지에 관한 연구동향

S/W의 유지·보수(maintenance)는 S/W 개발노력의 50-80%를 차지하는 과정이다. S/W 보수·유지활동의 효율성 극대화는 바로 S/W 개발팀의 생산성 제고로 이어진다[9, 10]. 이런 맥락에서 Alam et al.[11]는 컴퓨터 코드의 수정작업을 평가하기 위한 모니터링과정과 평가과정을 제시한 바 있으며, 유지보수작업에 대한 요구가 최근에 수행된 수정작업에 관한 것인지, 아니면 이전에 이루어졌던 작업과 연계가 되는지, 혹은 전혀 새로운 이슈에 대한 요구인지를 구분해야 할 필요성을 제기하였다. Henry et al.[12]는 S/W의 보수·유지작업을 활동(activities), 과업(tasks), 그리고 과정(procedures)의 3가지 개념적 차원으로 구분할 것을 제안하고 각 수준별 유지보수작업에 대한 평가지표를 논의한 바 있다. Juergens[13]는 많은 경우 이런 평가와 관련된 지표는 S/W 보수·유지활동의 효율성 제고에 큰 기여를 하지 못한다고 비판을 제기하기도 하였다.

다양한 환경과 다양한 고객의 요구를 충족시켜주다 보면 여러 S/W 모듈 간의 상호작용과정에서 오류나 문제점이 발생하게 된다. 개발시점에서는 고려하지 못했던

이슈나 요구조건이 제시될 수 있기 때문이다[14]. 따라서 오류의 수정이나 이슈제거는 S/W의 개발활동의 가장 큰 비중을 차지할 수밖에 없다. 따라서 이슈나 오류제거에 소요되는 시간의 추정은 개발팀의 효율성 제고에 필수적인 과정이다. 이슈나 오류제거에 소요되는 시간에 영향을 미치는 요인은 매우 다양하다. Kim et al. [15]은 ArgoUML과 PostgreSQL 프로젝트에서 오류(bugs)의 제거과정을 분석한 바 있는 평균 200일 정도 걸리는 것으로 보고한 바 있다. Giger et al.[16]도 3개의 오픈소스 프로젝트에 의사결정나무(Decision Tree) 기법을 적용하여 오류제거에 소요되는 시간을 추정한 바 있으며, Zhang et al.[14]은 마야코브 체인을 활용한 모형을 토대로 하여 추정가능한 오류의 개수와 오류수정에 소요되는 시간을 추정하는 방법을 제안한 바 있으며, 오류에 대한 정보가 주어지면 경우 오류제거 소요시간이 일정 수준 이상 길어질지 여부로 평가할 수 있는 방법도 제시하였다.

Petersen과 Wohlin[17]는 링경영 관점에서 S/W 보수·유지작업을 일종의 재고로 간주하여 다양하게 유형화하고 그 속성을 논의한 바 있다. 요구사항의 반영, 사례 검증, 수정 요청, 오류 수정 등으로 유형화하였다.

- 요구사항의 반영(requirements) - 사용자가 요구하는 요구사항에 대한 기능 포인트를 이용하여 요구사항의 경중을 평가한다.
- 사례 검증(test cases) - S/W의 성능 테스트에 요구되는 사례수, 검증의 복잡도, 테스트 담당자의 전문성 정도 등을 토대로 하여 투입되는 노력을 평가할 수 있다.
- 수정요청(change requests) - 수정요구사항이 S/W 운용에 미치는 영향력 정도를 감안하여 수정요청에 투입되는 노력을 평가할 수 있다.
- 오류 수정(faults): 오류수정에 소요되는 작업 범위와 난이도를 감안하여 투입되는 노력을 평가할 수 있다.

또한 각각의 작업은 다시 고차원 요구사항, 상세한 요구사항, 디자인 상의 요구사항, 출시 과정에서의 요구사항 등과 같이 여러 개가 결합된 상태를 정의할 필요가 있음을 주장하였다. 각각의 보수·유지작업을 일종의 재고로 봄으로써, 재고에 대한 지속적인 모니터링과 평가를 용이하게 하며 재고상태를 작업 중인지 또는 버퍼에

서 일시적으로 대기 중인지도 알 수 있도록 한다는 것이다. 또한 보수·유지작업 요구가 하나의 재고인지 여러 개의 재고로 구성되어 있는지도 알 수 있게 해주고 이들 요구의 상호작용이 가져다 줄 작업 자체의 난이도나 요구 품질유지에 대한 어려움을 이해하는데 도움을 준다.

## 4. 장비관련 소프트웨어 이슈 및 보수·유지에 관한 실증 분석

### 4.1 자료수집 및 변수 정의

본 연구의 분석자료는 반도체 장비회사 A사에서 2012년부터 2016년 사이에 반도체 웨이퍼 생산장비를 구동하는 S/W 관련 문제로 보고된 사건을 대상으로 502개의 데이터를 무작위 추출방식으로 수집하였다.

반도체 장비 S/W에 관련한 사안(issue) 발생 후부터 해결까지의 상황을 반영한다고 가정한 연구변수들은 다음과 같다. 먼저 독립변수로 설정된 SDR 변수는 'Software development request'를 의미하는 것으로, 사전에 제출된 S/W에 대한 개발 요청사항이 존재했는지 여부를 의미한다. SDR 변수값은 개발요청서가 제출은 되었으나, 해당 요청이 반려되거나, 중복, 또는 통합 즉 개발 완료 후 병합(integrated)된 상태일 수 있다. 따라서 제기된 S/W 관련 사안은 S/W 개발요청과 연관이 있을 수도 있고 없을 수도 있다.

또 다른 독립변수인 VD는 'Visual Description'을 의미하는 것으로 시각 자료가 사안이 공식적으로 제기된 초기시점에 첨부되어 함께 제시된 상황인지 여부이다. 사안의 해결과정에 추후 첨부되는 시각 정보는 본 연구에서 정의한 VD에 해당되지 않는다. 시각자료는 bmp, jpg, png 등의 확장자로 저장된 자료로서 일반적으로 UI화면의 캡처를 이용하여 논의 중인 사안과 연관된 S/W UI, 구동 프로그램 등에 관한 정보를 포함한다.

LOE는 사안에 대한 S/W 로그(log)정보의 포함 여부를 나타낸다. 로그정보는 S/W 작동시점부터 매 시점의 적용상태 및 상태의 변화를 시간정보와 함께 실시간으로 기록한 자료를 의미한다. 따라서 로그정보는 제기된 S/W 문제를 분석하여 해결하는데 기초자료로 활용될 수 있다. 다음으로 'Issue Type'는 이슈의 종류를 문제제기, 개선, 그리고 정보요청으로 구분한 이산형 변수이다. 구

체적으로 본 변수의 옵션을 살펴보면, 문제제기는 고객이나 회사 내부에서 반도체 장비S/W 나 그에 연관된 각종 문제점 등에 대하여 최초로 사안을 발의한 상황이다. S/W 개발(개선)요청은 개발자가 올바르게 개발할 수 있도록 요청사항을 주로 개선에 초점을 맞추어 명확히 작성하여 제출한 상황이다. 정보요청은 주로 문제는 아니고, 궁금한 사항 혹은 별도의 정보가 필요할 시 청구된 상황이다. 정보요청에는 고객뿐만 아니라 조직 내부적으로 제출된 질의사항도 포함된다.

DWT은 'Department Work Type'를 의미하며 열처리 (rapid thermal processing), 증착(deposition), 금속화 (metalization), CMP(chemical mechanical polishing), 식각(etching) 등 5가지 공정 중 제기된 사안의 담당 사업부(혹은 담당부서)를 의미한다. 각 회사별로 취급하는 반도체 공정 장비는 좀 더 세분화 되어 있고 공정의 순서나 종류는 서로 다를 수 있다. 그러나 반도체 공정의 기본적인 진행 순서를 근간으로 본 연구에서는 장비가 실질적으로 나누어진 5가지의 공정 사업부로 정리하였다. 이외에도 별도 반도체 공정 장비 사업부가 있으나 S/W 사안 관리 방식의 차이(데이터베이스, 처리방식 차이 등)로 인해 제외하였다.

중속변수로 설정한 DR은 'Issue Duration', 즉 이슈 해결기간을 의미하며 S/W관련 사안이 제출된 시점부터 최종적으로 사안이 종결된 시점 사이의 시간을 의미한다. 사안의 종결은 종료 권한을 가진 관리자가 판단한다.

CMNB변수는 해당 사안에 관한 커뮤니케이션 횟수를 의미한다. 여기서 커뮤니케이션 횟수는 최초 S/W관련 사안의 제기 후 문제 해결을 위해 DB에 기입한 견해 (comment) 건수로 측정한다. 견해는 글로 쓰인 정보, 시각정보 등 다양한 자료를 삽입한 설명으로, 이슈를 제기한 당사자와 해결을 위한 제3자 모두가 등록할 수 있다. 연구변수에 대한 요약은 <Table 1>과 같다.

#### 4.2 변수의 분포에 대한 가정

실제 작업환경에서 얻은 본 연구의 자료인 사안해결 기간이나 의견교환 횟수의 분포는 모두 정규분포와 다를 가능성이 높다. 품질경영분야에서는 문제발생의 주요 원인인 이상원인(assignable causes)과 같이 쉽게 확인되고 해결되는 부분에서 주로 발생한다고 본다. 이러한 맥락에 따라, 본 연구의 분석에 사용되는 변수인 해결기간이나 의견교환 횟수는 모집단의 상당수가 짧은 기간이나 적은 횟수에 집중될 것으로 기대된다. 또한 전자는 연속형 변수, 후자는 범주형 가산변수이고 두 가지 모두 양의 값을 갖는 특징이 있다.

선형회귀모형을 이용하여 이 같은 가산자료를 분석할 경우, 잔차의 이분산성 및 비정규성으로 인하여 추정 확률에 대한 유의성이 결여되기 쉽다[18]. 또한 중속변수가 연속형이라도 정규분포를 가정할 수 없다면 선형회귀모형의 상정은 적절하지 않다. 이러한 추정과정 상의 문제 해결도구로서 제단이나 사고의 발생확률 등에 관련된 여러 연구에서는 감마 회귀모형과 포아송 회귀모형 그리고 음이항 회귀모형 등의 일반화 선형모형(generalized linear model; GLM)을 광범위하게 사용하고 있다.

일반화 선형모형은 중속변수와 설명변수 사이의 관계를 평가하고 정량화한다는 점에서는 선형회귀모형과 같으나, 일반화 선형모형은 분석을 위해 중속변수의 분포를 정규분포에 한정하지 않고, 이항, 감마, 역가우스, 음이항(negative binomial), 포아송 분포 등 다양한 분포가 선택될 수 있다. 또한 정규선형모형에서는 중속변수와 설명변수 사이의 연결함수 관계속성이 항등함수인 것에 반하여(e.g.,  $y = \beta x$ ), 일반화 선형모형에서는 중속변수가 특정 연결함수를 통해 변환되며, 이 값이 설명변수와 선형적으로 관계가 있다(e.g.,  $g(y) = \beta x$ ). 포아송 분포는 발생횟수가 0, 1, 2 혹은 3 등의 이산적인 분포형태인 사건의 발생확률 연구에 적용되고 있다. 중속변수의 분

<Table 1> Definition of Research Variable

Variable Name	definition
SDR	Software development request No : 0, , Yes : 1
VD	Visual description No : 0, , Yes : 1
LOE	log data existence concerning problem issues No : 0, , Yes : 1
TY	Issue type 1 : issue raising, 2 : improvement 3 : information requests
DT	Work Center Type 1: RTP, 2: deposition, 3: metalization, 4: etching, 5: CMP
DR	Issue duration
CMNB	The number of information or comments exchange

포를 포아송 분포로 가정한 포아송 회귀모형은 종속변수의 평균과 분산이 같다는 가정을 전제한다. 과대산포(overdispersion)가 존재하면 회귀계수의 표준오차가 편향되는 결과를 초래하여 추정결과의 신뢰성이 저하되기 때문이다[19].

이러한 맥락에서, 포아송 모형에 대한 대안으로 음이항 회귀모형(negative binomial regression model)을 활용하기도 한다[20, 21]. 음이항 회귀모형은 포아송 분포와 감마 분포의 혼합으로 생성된 음이항 분포를 사용하며, 포아송 회귀모형과 유사한 함수적 형태를 갖지만 과대산포를 반영한 추정회귀계수를 구할 수 있다[21].

감마 분포는 포아송 분포이나 음이항 분포와 유사한 분포 모양을 보인다. 큰 차이점은 포아송 분포와 음이항 분포는 계수(count)가 가능한 빈도의 이산확률인 반면, 감마 분포는 연속확률이라는 점이다. 감마 회귀모형은 감마분포를 따르는 종속변수가 연결함수로 조건화되어 설명변수와 선형적으로 관계를 갖는다고 가정한 것이다. 이러한 특징들은 일반화 선형모형이 자료의 정규성이 결여될 것으로 생각되는 본 연구의 자료분석을 위한 적절한 방법론임을 알려준다.

앞서 언급한 바와 같이, 본 연구의 자료인 사안해결기간이나 의견교환 횟수는 모두 정규분포를 따르지 않으며 과대산포가 존재할 가능성이 높다. 다시 말해, 본 연구의 실증자료는 정규분포를 가정한 모형분석에는 적합하지 않다. 또한, 종속변수에 영향을 줄 것으로 추정된 설명변수는 모두 명목형 특징을 갖는다. 이를 고려하여 본 연구에서는 이러한 조건을 수용할 수 있는 분석방법론으로 활용되고 있는 감마 회귀모형과, 포아송 회귀모형 또는 음이항 회귀모형을 적용하고자 한다.

#### 4.4 분석결과 및 해석

실증분석에 앞서, 표본자료구조 탐색을 위하여 종속변수인 이슈해결기간과 커뮤니케이션 횟수를 살펴본 결과는 이슈해결기간의 확률분포는 정규분포를 따르지 않는 것을 확인할 수 있었으며, 커뮤니케이션 횟수 역시 정규분포가 아닌 모습의 분포가 나타남을 확인할 수 있다.

이에 본 연구는 이슈해결기간의 분포는 감마분포, 그리고 커뮤니케이션 횟수는 음이항분포를 따른다고 가정한 후, 상정한 일반화 선형모형의 모형적합지수를 통해 종속변수의 분포적합여부와 상정모형의 타당성을 함께

확인하였다. 기본적으로 상정한 모형의 분석타당성을 확인하기 위해서는 일반화 선형모형에서는 모형전체의 우도비 카이제곱검정을 시행한다. 이와 함께 '총 이탈도'(deviance/df)도 확인하는데, 일반적으로 이 값이 1에 근접할수록 관측데이터가 적용한 회귀모형에 적절하다고 고려하며, 4보다 클 경우에는 모형이 적합하지 않다고 본다[21]. 우선 종속변수로 이슈해결기간을 설정하고 다음과 같은 감마회귀모형(gamma regression model)을 분석하였다.

$$y \sim G(\mu, \nu); \ln \mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5$$

분석결과인 <Table 2>를 살펴보면, 모형 자체가 무의미하다는 귀무가설(모든 회계계수값이 0인 것으로 가정)을 우도비(likelihood ratio)  $\chi^2$ 을 이용하여 검정한 결과 유의수준 1%에서 기각되었으며( $\chi^2 = 46.833, p < 0.001$ ), 총 이탈도도 1.251로서 1에 근접하므로, 감마회귀모형의 적용은 적절한 것으로 분석되었다[22].

<Table 2> The result of Gamma regression analysis (Dependent Variable : Issue Duration)

	$\hat{\beta}$	SE	Wald $\chi^2$
intercept	3.789***	.2767	187.613
SDR (1)	.389*	.1605	5.879
(0)	0		
VD (1)	.100	.1152	.749
(0)	0		
LOE (1)	.427**	.1353	9.954
(0)	0		
type			14.055***
TY Information requests	-.936***	.2733	11.741
improvement	0		
issue raising	-1.015***	.2714	13.973
work center			2.464
CMP	.235	.1807	1.690
DT etching	.222	.1726	1.651
metalization	0		
deposition	.131	.1760	.554
RTP	.111	.1615	.469
$\hat{\nu}$	1.577	.0840	
Likelihood ratio $\chi^2 = 46.833$ ***			
Log likelihood = -2155.094			
deviance/df = 1.251			

\*  $p < .05$ , \*\*  $p < .01$ , \*\*\*  $p < .001$

이슈해결기간 기댓값의 변화에 영향을 미치는 독립변수로는 SDR(  $Wald \chi^2 = 5.879, p = .015$ ), LOE(  $Wald \chi^2$

=9.954, p=.002), 그리고 TY( $Wald\chi^2=14.055$ , p=.001)로 유의한 것으로 나타났으며, VD( $Wald\chi^2=.749$ , p=.387)와 DT( $Wald\chi^2=2.464$ , p=.651)는 이슈해결기간 기댓값 변화를 설명하는데 기여하지 못하는 것으로 나타났다.

유의한 설명변수의 영향을 구체적 살펴보면, SDR의 경우 SDR = 1인 경우 ( $\hat{\beta}=.389$ , p=.015), 즉 다른 환경요인이 동일할 경우, 이슈 해결기간의 기댓값( $\mu$ )은 SDR가 '없음'에 비하여 SDR가 '존재'할 때  $e^{.389}(=1.476)$ 배 만큼 증가된다고 볼 수 있다. 로그정보 LOE도 값이 1인 경우 ( $\hat{\beta}=.427$ , p=.002) 이슈해결기간이 길어진다. 여타 조건이 동일할 경우, 이슈 해결기간의 기댓값( $\mu$ )이 LOE가 '없음'보다 LOE가 '존재'할 때  $e^{.427}(=1.532)$ 배만큼 커진다.

TY는 개선(improvement)을 기준으로 사안발의(issue raising)( $\hat{\beta}=-1.015$ , p=.000)와 정보요청(information request)( $\hat{\beta}=-.936$ , p=.001)에서 이슈해결기간이 유의미하게 짧다고 볼 수 있으며, 정보요청과 사안발의 사이의 이슈해결기간의 기댓값 차이는 대비 검정결과 무의미하다(p=.563). 따라서 여타 조건이 동일하다면 이슈해결기간의 기댓값( $\mu$ )은 개선일 경우 대비 사안발의일 때는 63.7%만큼 짧고 정보요청일 때는 60.8% 짧음을 의미한다.

한편, 기업 내 문제해결상황을 간접적으로 살펴볼 수 있는 또 다른 지표로 선정된 커뮤니케이션 횟수를 종속변수로 설정하고 새로운 분석모형을 통해 탐색을 시도하였다. 우선 커뮤니케이션 횟수의 과대산포 존재여부를 통계적으로 검정하였는데, 과대산포의 존재 유무에 대한 가설검정은  $H_0: \kappa = 0$ 이라는 귀무가설을 우도비(likelihood ratio) 검정하는 것이다.  $\kappa$ 는 과대산포모수(overdispersion parameter)이다. 만일, 귀무가설이 기각된다면 포아송모형이 아닌 음이항 회귀모형을 사용하는 것이 적절하다고 볼 수 있다. 귀무가설에 대한 우도비 검정통계량을 계산한 결과  $likelihood\ ratio\chi^2 = 209.452$  ( $df = 1$ )로 유의수준 1%에서 기각되어 종속변수에 과대산포가 존재하는 것으로 판단되어, 다음과 같은 음이항 회귀모형(negative binomial regression model)을 이용하는 것으로 결정하였다.

$$y \sim NB(\mu, \kappa); \ln \mu = \gamma_0 + \gamma_1 x_1 + \gamma_2 x_2 + \gamma_3 x_3 + \gamma_4 x_4 + \gamma_5 x_5$$

우선 이 모형을 적용한 결과를 요약한 <Table 3>을 살펴보면, 모형 자체가 무의미하다는 귀무가설(모든 회귀계수값이 0인 것으로 가정)을 우도비 $\chi^2$ 을 이용하여 검정한 결과  $likelihood\ ratio\chi^2 = 43.973$ 으로 유의수준 0.1%에서 기각되었다. 또한 총 이탈도는 1.005로 1에 매우 근접하므로, 상정한 음이항 회귀모형분석의 결과를 활용할 수 있다[20]. 종속변수인 커뮤니케이션 횟수의 변화에 영향을 줄 것으로 예상한 설명변수의 통계적 유의성을 살펴보면, SDR( $Wald\chi^2=6.643$ , p=0.010), LOE( $Wald\chi^2=15.907$ , p=0.000), 그리고 TY( $Wald\chi^2=6.714$ , p=.035) 등이 유의미하게 확인되었다. VD와 DT는 모두 p값이 5%를 초과하여 통계적으로 유의하지 않은 것으로 나타났다.

<Table 3> The result of Negative binomial regression analysis(Dependent Variable : the number of communication)

	$\hat{\beta}$	SE	Wald $\chi^2$
intercept	.442*	.1995	4.916
SDR (1)	.256**	.0991	6.643
(0)	0		
VD (1)	.017	.0730	.053
(0)	0		
LOE (1)	.352***	.0881	15.907
(0)	0		
TY type			6.714*
Information requests	.496*	.1953	6.448
improvement	0		
issue raising	.481*	.1919	6.290
work center			8.237*
DT CMP	.084	.1131	.546
etching	-.270*	.1144	5.590
metalization	0		
deposition	-.046	.1106	.171
RTP	-.031	.1023	.091
$\kappa$	.331	.0394	

Likelihood rate  $\chi^2 = 43.973$ \*\*\*

Log likelihood = -1090.731

deviance/df = 1.005

overdispersion parameter( $\kappa$ )

\*  $p < .05$ , \*\*  $p < .01$ , \*\*\*  $p < .001$ , +  $p < .1$

유의미한 설명변수의 영향을 구체적 살펴보면, SDR = 1인 경우 SDR = 0인 경우 보다 커뮤니케이션 횟수가 더 많은 것으로 나타났으며( $\hat{\beta}=.256$ , p=.010), 로그정보 LOE도 LOE=1인 경우가 0인 경우보다 커뮤니케이션 횟



수가 많다( $\hat{\beta}=.352, p=.000$ ). 즉 다른 환경요인이 동일할 경우, 커뮤니케이션 횟수의 기댓값은 SDR가 '없음'에 비하여 SDR가 '존재'할 때  $e^{.256}$ 배 즉, 29.1% 증가되며, LOE가 '없음'보다 LOE가 '존재'할 때  $e^{.352}$ (= 1.431) 배 만큼 증가하는 것으로 분석되었다.

TY는 개선(improvement)의 경우 기준으로 할 때 사안발의( $\hat{\beta}=.481, p=.012$ )와 정보요청( $\hat{\beta}=.496, p=.011$ )의 경우 상대적으로 커뮤니케이션 횟수가 유의미하게 증가한다고 볼 수 있으며, 정보요청과 사안발의 사이의 커뮤니케이션 횟수 기댓값 차이는 유의미하지 않은 것으로 나타났다( $p=.868$ ). 따라서 여타 조건이 동일하다면 의견교환횟수는 개선일 경우 대비, 사안발의일 때는 64.2% 많고 정보요청일 때는 61.8% 많아질 것으로 예측된다.

위의 분석결과를 종합하면 다음과 같이 해석해 볼 수 있다. 사안 발생 전에 기 제출된 S/W 개발 요청사항(SDR)이 존재한다면 해당 사안의 해결기간은 증가하게 된다. 통상적으로 S/W 개발기간은 중요도, 난이도, 하드웨어 접합 여부, 계획 등 개발과정의 복잡성에 따라 충분히 길어질 수 있다. 그러므로 새로운 S/W가 장비운영에 적용되는데 까지 걸리는 시간은 길다. 만일 S/W 개발 요청사항이 존재했던 상황이라면 S/W 개발과정이 보다 복잡해질 것으로 예상되므로, 이러한 복잡한 상황에서는 제기된 사안에 대한 검토 시간뿐만 아니라 해결과정에도 추가적인 시간이 소요되는 것으로 추정할 수 있다. 더불어 S/W 개발 프로세스가 복잡해진 만큼 검토사항이 증가될 것이므로 사안 해결을 위해 고객 혹은 조직내부 사이의 의견교환 횟수가 증가되는 것으로 해석할 수 있다.

로그정보는 일종의 이정표 같은 속성으로 제기된 사안이나 문제의 원인을 빠르게 진단하는데 도움이 될 것으로 예상하여 사안의 해결기간을 단축시킬 것으로 기대하였다. 그러나 통계자료의 분석결과는 기대와 상반된다. 사안의 해결기간과 의견교환 횟수의 증가는 로그정보가 포함된 사안일수록 사안의 복잡성이나 해결 난이도가 증가했기 때문에 나타나는 현상으로 이해해 볼 수 있다.

사안의 종류 중 개선과정에서 커뮤니케이션 횟수는 줄고 해결기간은 늘어난 것은 사안이나 문제점 자체를 이해하기 위한 상호간의 의견교환이 상대적으로 덜 필요하기 때문이며, 오히려 S/W 자체의 기술문제를 해결하는 내부작업 과정이기에 해결기간이 긴 것은 당연하다.

또한 문제제기나 정보교환 과정에서 상대적으로 커뮤니케이션 횟수가 많다는 것은 조직 내부나 고객과 서로 문제에 대한 접근과 이해를 같이 하기 위해, 용어를 정의 및 공유하고 정보와 지식수준을 일치시키는데 조직역량을 투입한다는 것으로 일반화해볼 수 있다. 이는 문제해결을 정확하고 조속히 진행하기 위해서 필수적인 행위이지만, 조직운영 상의 소요되는 외부시간을 증가시키므로 전체적으로는 사안의 해결기간이 증가한다고 생각해볼 수 있다.

한편, 시각적 정보(VD)가 포함되면 사안의 해결기간과 커뮤니케이션 횟수를 감소시킬 것으로 예상하였다. 즉, 시각정보의 활용으로 제기된 사안에 대한 설명은 직관적이지만 단번에 보다 상세한 정보를 제공하게 될 것으로 예상하여, 의견교환 횟수의 감소와 사안의 조속한 해결에 도움이 될 것이라 생각했다. 그러나 분석결과는 시각정보의 활용유무는 사안의 해결기간을 단축시키거나 사안에 대한 의견교환 횟수를 증감시키는 결정적인 요인이라고 보기는 어려운 것으로 나타났다.

더불어 현재 이 기업은 사업부 단위별로 사안해결기간의 편차가 없으며 커뮤니케이션 횟수에서도 편차의 존재는 통계적으로 뚜렷하다고 보기 어렵다. 결과적으로 현재 기업 내 S/W개발조직직원의 배분상황과 사업부별 해결역량은 비슷하다고 평가해볼 수 있다.

## 5. 결론

반도체 산업의 규모가 한국 경제에서 차지하는 비중을 고려할 때 거시적 차원의 관심과 더불어, 산업군 내 개별기업에 대한 경영학적 접근이 함께 진행되어야 할 시기이다. 반도체 웨이퍼 제조공정의 장비운영 S/W처럼, 산업현장에서 사용되는 복잡한 알고리즘의 S/W의 유지보수는 산출제품의 품질이나 생산경제성에 직접적으로 영향을 미치는 중요한 과정이다. S/W에 문제가 발생한다면 신속한 대처를 통해 장비의 생산성 감소폭을 줄여야 한다. 그러나 제기된 문제나 사안의 해결에는 공학기술적인 측면뿐만 아니라 운영관리 차원의 지원미비 등으로 신속한 진행이 어려운 경우가 많다.

기업의 문제해결역량은 경험의 축적에 비례한다. 그러나 제조장비라는 산업재의 특성 상, 반도체 웨이퍼 제조

장비 임베디드 S/W의 개발 혹은 유지보수는 소비재의 임베디드 S/W분야에 비하여 개별기업이 축적할 수 있는 경험의 기회가 적다.

이러한 시각의 연장에서 본 연구는 운영 관리적 측면에서 제기된 사안이나 문제를 조속히 해결하기 위해서는 어떠한 해결과정 조건이나 환경을 유도/준비해야 하는지에 관한 함의를 간접적으로 제공하는데 기여한다고 평가할 수 있다. 우선 실무처리 전반의 일정관리나 프로젝트를 관리하는 일선관리자의 경우, 제기된 사안의 S/W개발 상황이나 조건을 확인하여 특정 조건을 보이는 사안들의 해결과정을 중점적으로 관리한다면 이슈해결기간을 보다 효율적으로 관리할 수 있으리라 본다. 즉, 사전 S/W에 관한 개발요청사항 혹은 로그정보가 포함된 사안이거나 제기된 사안의 종류가 현재 S/W에 대한 개선요청이라면 사안의 해결과정에 집중적인 관심을 갖고 관리하여 보다 효율적인 일정관리를 할 수 있을 것으로 기대한다. 이와 함께 본 연구의 결과를 해결기간이 길 것으로 예상되는 S/W 관련 사안을 선정하는 기준으로 활용한다면, 전사적으로는 해당 사안에 조직역량이나 자원을 집중시켜 과도하게 해결기간이 길어지는 것을 방지하는데 도움이 될 것으로 기대한다.

이러한 활용가능성에도 불구하고, 실무 입장에서는 본 연구 내 각 변수의 효과해석이나 일부 변수의 효과가 각각된 점을 직관적으로 납득하기 어려운 부분이 있을 수 있다. 이러한 의문사항은 제기된 이슈의 해결과정이나 상황을 설명해주는 변수의 추가와 데이터의 양적 확대 후 통계적 재검증을 통해 해소될 것으로 기대한다.

## REFERENCES

- [1] S. Kang, H. Yang, "Quality Evaluation of Criterion Construction for Open Source Software." *Journal of Digital Convergence*, Vol. 11, No. 2, pp. 323-330, 2013.
- [2] J. Hong, "Security Policy Issue in Application Software Development Process of Smart Phone Environment." *Journal of Digital Convergence*, Vol. 10, No. 10, pp. 319-324, 2012.
- [3] A. Kim, J. Roh, S. Rhee, "Implementation of an Expert System for COTS Fault Diagnosis." *Journal of Digital Convergence*, Vol. 11, No. 1, pp. 275-281, 2013.
- [4] Y. Lee, "Review of the Embedded Software status of Korea for Semiconductor", *Communications of the Korean Institute of Information Scientists and Engineer*, Vol. 28, No. 10, pp. 10-15, 2010
- [5] H. Kwon, "Plan for training the competent human resources on Embedded SW", *Magazine of the IEEK*, Vol. 39, No. 1, pp. 49-53, 2012.
- [6] H. Yoo, J. Lee, H. Kim, "An analysis on the Secotral Innovation System of Embedded Software industry in Korea." *Proceedings of Symposium of the Korean Institute of communications and Information Sciences*, pp. 47-48, 2012. 6.
- [7] H. Yang, J. Moon, M. Jeong, "Analysis of Embedded SW Technology And Strategy of Industrial Foundation Competence." *The Journal of The Korean Institute of Communication Sciences*, Vol. 35, No. 8, pp. 1232-1240, 2010.
- [8] Sang-Yong Kim and Byeong-Cheol Lee, *Semiconductor CMOS manufacturing technique*, Seoul: Iljinsa, 2016.
- [9] K. Petersen, "A palette of lean indicators to detect waste in software maintenance: A case study", In: *Proceedings of 13th International Conference on Agile Processes in Software Engineering and Extreme Programming*, pp. 108-122, 2012.
- [10] K. Petersen, "Is lean agile and agile lean? A comparison between two software development paradigms", In: Dogru, A. H. and Bicer, V. (ed). *Modern Software Engineering Concepts and Practices: Advanced Approaches*, pp. 19-46, 2010.
- [11] O. Alam, B. Adams, and A. E. Hassan, "Measuring the progress of projects using the time dependence of code changes", In: *Proceedings of 2009 IEEE International Conference on Software Maintenance*, pp. 329-338, 2009.
- [12] J. Henry, R. Blasewitz, and D. Kettinger, "Defining and implementing a measurement-based software maintenance process." *Journal of Software*

Maintenance and Evolution, Vol. 8, No. 2, pp. 79-100, 1996.

- [13] J. Elmar, "Software maintenance research that is empirically valid and useful in practice." *it-Information Technology*, Vol. 58, No. 3, pp. 145-149, 2016.
- [14] H. Zhang, L. Gong, and S. Versteeg, "Predicting bug-fixing time: An empirical study of commercial software projects", In: *Proceedings of the 35th International Conference on Software Engineering (ICSE)*, pp. 1042-1051, 2013.
- [15] S. Kim, and Jr ,E. J. Whitehead, "How Long Did It Take to Fix Bugs?", In: *Proceedings of the 2006 International Workshop on Mining Software Repositories*, pp. 173-174, 2006.
- [16] E. Giger, M. Pinzger, and H. Gall, "Predicting the Fix Time of Bugs", In: *Proceedings of the 2nd International Workshop on Recommendation Systems for Software Engineering*, pp.52-56, 2010.
- [17] K. Petersen and C. Wohlin, "Software Process Improvement through the Lean Measurement (SPI-LEAM) Method." *Journal of Systems and Software*, Vol. 8, No. 7, pp. 1275-1287, 2010.
- [18] R. Blundell, R. Griffith, and J. Van Reenen, "Dynamic Count Data Models of Technoical Innovation." *The Economic Journal*, Vol. 105, No. 429, pp. 333-344, 1995.
- [19] W. Greene, "Functional forms for the negative binomial model for count data." *Economics Letters*, Vol. 99, No. 3, pp. 585-590, 2008.
- [20] R. Baptista, and P. Swann, "Do firms in clusters innovate more?" *Research Policy*, Vol. 27, No. 5, pp. 525-540, 1998.
- [21] P. Jong and G. Z. Heller, *Generalized linear models for insurance data*. New york: Cambridge University Press, 2008.
- [22] J. M. Hilbe, *Negative binomial regression*. 2nd ed. Cambridge, UK: Cambridge University Press, 2011.

조 남 형(Cho, Namhyung)



- 2010년 2월 : 아주대학교 생명과학/경제학(이학사/경제학사)
- 2012년 8월 : 성균관대학교 MS in management(경영학석사)
- 2015년 8월 : 성균관대학교 경영전문대학원(경영학 박사)
- 관심분야 : 공급망관리, 디지털 커뮤니케이션

· E-Mail : namhyung.cho@gmail.com

김 치 린(Kim, Chi Rin)



- 2010년 2월 : 경상대학교 산업정보공학과(공학학사)
- 2017년 2월 : 성균관대학교 경영대학원 IMBA (경영학석사)
- 2010년 7월 ~ 현재 : 어플라이드 머티어리얼즈 코리아(S/W 기술지원, 자동화)
- 관심분야 : 반도체 자동화

· E-Mail : rinchi1013@gmail.com

김 미 량(Kim, Mi Ryang)



- 1987년 2월 : 서울대학교 인문대학 영어영문학과(B.A.)
- 1988년 10월 : Lehigh University, Edu. Tech. (M.S.)
- 1998년 2월 : 서울대학교 교육학과 교육방법/공학 (Ph.D.)
- 1999년 3월 ~ 현재 : 성균관대학교 컴퓨터교육과 교수

· 관심분야 : SW관리/교육

· E-Mail : mrkim@skku.ac.kr