

임베디드 시스템 개발 단계별 설계 결정에 대한 추적성 연구

최강식⁰ 문미경 영근혁

*부산대학교 컴퓨터공학과

{siki75⁰, mkmoon, yeom}@pusan.ac.kr

An Traceability Research for Design Decision in the Embedded System Development Life Cycle

Kangsik Choi, Mikyeong Moon, and Keunhyuk Yeom

Department of Computer Engineering , Pusan National University

요 약

임베디드 시스템은 특정 목적을 수행하기 위해 개발되는 컴퓨팅 시스템으로 최적화된 하드웨어 위에서 동작 가능한 소프트웨어로 구성된다. 임베디드 시스템은 개발 단계별로 여러 가지 설계 대안 중 최적의 설계를 선택하는 설계 결정이 이루어지는데, 설계 결정 시에는 이에 대한 근거가 제시되어야 한다. 임베디드 시스템은 여러 유형의 제약사항 범위 안에서 개발되어야 하는 특징으로 인해 임베디드 시스템 개발 시에는 기능적 요구사항과 함께 비기능적 요구사항 역시 중요한 요소가 된다. 따라서 제약사항을 담고 있는 비기능적 요구사항은 설계 결정 시에 중요한 영향을 미친다. 그러므로 본 논문에서는 CMMI의 공학 프로세스 영역에 근거하여 임베디드 시스템의 개발주기를 요구사항 개발, 기술적 해결, 제품 통합 각 단계로 나눈 다음, 각 단계에서 이루어지는 설계 결정 활동을 비기능적 요구사항에 근거하여 분석하고자 한다. 이를 통해 임베디드 시스템 개발 단계별 설계 결정에 대한 추적성을 제공하고자 한다.

1. 서 론

최근 유비쿼터스 컴퓨팅 분야가 이슈로 떠오르면서 임베디드 시스템에 대한 관심이 증대되고 있다. 임베디드 시스템은 일반 데스크탑 기반 시스템과 많은 차이점을 가지고 있다. 임베디드 시스템은 특정 목적을 위해 개발되어지고, 하드웨어 자원의 제약을 많이 받는 시스템이다[1]. 하드웨어와 소프트웨어를 모두 개발해야 하며, 소프트웨어는 특정 목적을 위해 설계된 하드웨어 위에서 동작하게 된다[2]. 따라서 임베디드 시스템의 경우 주어진 여러 유형의 자원 제약사항 (크기, 메모리, 마이크로 프로세스, 전력 소모 등) 을 고려한 개발이 이루어져야 한다[3].

현재 임베디드 시스템에 대한 연구는 SoC (System-on-Chip)를 위한 실리콘 패키징 기술, 범용 개발 환경 구축 기술 및 운영체제, 미들웨어 개발 기술 등의 구현 기술에 대부분의 연구가 집중되어 있다. 임베디드 시스템의 개발 생산성을 향상시킬 수 있는 개발 방법 및 개발 단계에 대한 연구는 상대적으로 미비한 실정이다[4]. 임베디드 시스템은 여러 가지 제약사항을 만족하면서 시스템의 목적된 기능을 수행해야 한다. 따라서 시스템의 구조 결정 시에 비기능적 요구사항이 많은 영향을 미치게 된다[5].

본 논문에서는 CMMI(Capability Maturity Model Integration)의 공학 프로세스(Engineering Process Area) 영역에 근거하여 임베디드 시스템의 개발주기를 요구사항 개발(Requirement Development : RD), 기술적

해결(Technical Solution : TS), 제품 통합(Product Integration : PI) 각 단계로 나눈 다음, 각 단계에서 이루어지는 설계 결정 활동을 비기능적 요구사항에 근거하여 분석하고자 한다. 이를 통해 임베디드 시스템 개발 단계별 설계 결정에 대한 추적성을 제공하고자 한다[6].

2. 관련연구

2.1 ROPES (Rapid Object-oriented Process for Embedded Systems)

ROPES는 UML을 기반으로 한 임베디드 시스템 개발 방법이다[7]. 그림 1은 ROPES의 반복적인 생명주기를 나타낸다. ROPES는 Analysis, Design, Implementation, Testing의 4단계를 반복적으로 수행하며 각각의 반복마다 Prototyping 개발을 통해 시스템을 검증하는 것이 특징이다. ROPES는 일반적인 개발 프로세스를 임베디드 시스템에 초점을 두고 기술하고 있으며, 각 단계별 설계 대안과 결정, 그리고 결정에 대한 추적성에 대한 설명은 부족하다.

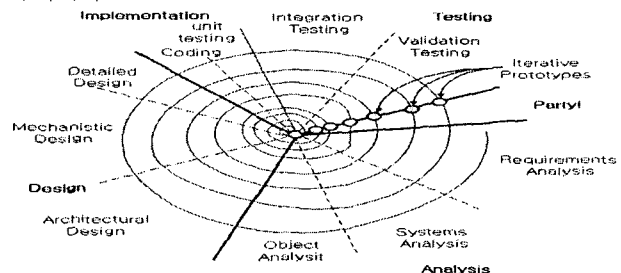


그림 1 ROPES의 반복적인 생명주기

본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성, 지원사업의 연구결과로 수행되었음

2.2 DAL (Decision Analysis and Resolution)

DAL은 CMMI의 지원(Support) 프로세스 영역의 하나이다[8]. DAL은 식별된 대안들을 수립된 기준으로 평가하고 결정사항을 분석한 다음, 수립된 기준을 바탕으로 대안들을 평가하여 최종의 해결안을 결정하는 가이드를 제시한다. DAL은 모든 프로세스 영역들에서 사용될 수 있도록 가이드를 제공하고 있기 때문에 임베디드 시스템에 특화된 가이드를 얻기가 부족하다.

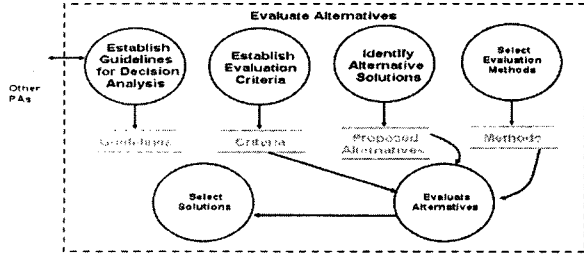


그림 2 DAL Process Area Context Diagram

3. 개발 단계별 설계 결정

3.1 Overview

임베디드 시스템의 개발 단계는 크게 요구사항 개발, 기술적 해결, 제품 통합으로 이루어진다. 그림 3은 각 개발 단계에서 이루어지는 설계 결정과 비기능적 요구사항의 관계를 보여주고 있다.

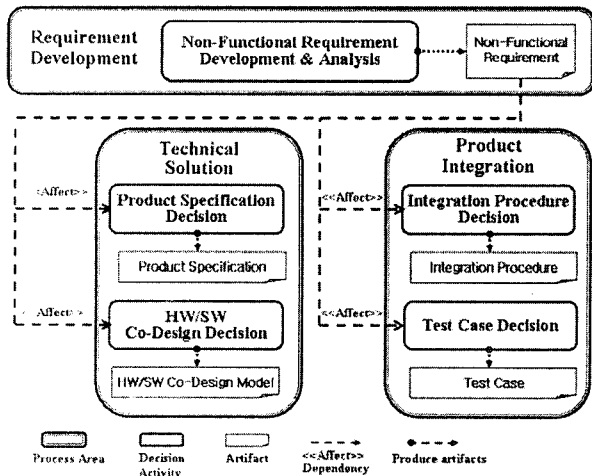


그림 3 개발 단계별 설계 결정 추적도

요구사항 개발 단계에서는 고객으로부터 고객이 원하는 요구사항을 수집하여 이를 만족하는 제품 요구사항을 개발하는 활동을 수행한다.

기술적 해결 단계에서는 개발된 고객 및 제품 요구사항을 토대로 제품에 대한 여러 가지 솔루션을 파악하고 정의된 선택 기준에 따라서 최적의 솔루션을 선택하게 된다. 최종 솔루션의 선택 후 상세 설계 및 구현이 이루어진다.

제품 통합 단계는 구현된 제품 구성요소들을 통합하는 단계로 통합 순서 및 통합 절차를 정의한다. 절차에 따른 시스템의 통합 및 제품 인도를 통해 임베디드 시스

템은 고객에게 전달된다.

3.1 요구사항 개발 (Requirement Development)

요구사항 개발 단계에서는 개발하고자 하는 임베디드 시스템에 대한 요구사항이 분석된다. 요구사항은 크게 기능적, 비기능적 요구사항으로 나눌 수 있다. 비기능적 요구사항은 제품의 기능 외 모든 요구사항을 의미하는 것으로, 크게 품질(Quality), 외부 인터페이스(Interface), 설계와 구현상의 제약사항(Constraint)으로 나눌 수 있다. 품질은 성능(Performance), 보안(Security), 실시간성(Real-Time) 등 시스템 마다 그 중요도가 다를 수 있는 특성들이다. 외부 인터페이스는 시스템이 접촉하는 하드웨어와 소프트웨어에 대한 인터페이스 제약사항을 의미한다. 설계 및 구현 제약사항은 반드시 사용해야 하는 개발언어, 데이터베이스, 기술 표준 등에 대한 제약사항을 의미한다. 요구사항 개발 단계에서는 최종적으로 만들어질 제품에 대한 명세를 작성한다. 시스템이 어떻게 동작해야 하는지, 시스템 속성이나 특성을 개발하여 문서화한다. 임베디드 시스템의 경우 비기능적 요구사항이 시스템 개발 단계 전체에 제약사항으로 영향을 주기 때문에 개발된 요구사항에 대한 상세한 분석이 필요하다. 본 논문에서는 임베디드 시스템의 특성을 고려한 비기능적 요구사항 분석 템플릿을 표 1에 제시한다. 이는 IEEE 830 표준에서 채택된 SRS(Software Requirements Specification) 템플릿을 바탕으로 비기능적 요구사항 항목을 상세화한 것이다[5].

표 1 비기능적 요구사항 분석 템플릿

품질	Name	Value
관련 품질요소를 정의	우선순위	
인터페이스		
하드웨어		
관련 하드웨어 인터페이스 정의		
소프트웨어		
관련 소프트웨어 인터페이스 정의		
제약사항		
관련 제약사항 정의		

표 2는 표 1의 템플릿을 이용한 임베디드 시스템의 비기능적 요구사항의 분석한 예의 일부분이다.

표 2 분석된 비기능적 요구사항

품질	Name	Value
성능	우선순위	2
	반응시간	5 sec
가용성	우선순위	1
	입출력처리시간	1 sec
	평균 동작 시간	24 hour
	복구시간	1 hour
인터페이스		
하드웨어		
입출력 시그널 인터페이스	입력	Digital
	출력	Digital
소프트웨어		
RTOS 와의 인터페이스	API	RTOS API
	Data 할당	빅 인디안
제약사항		
시스템 대역폭	FSB버스대역폭	128KB이상
	메모리대역폭	256MB/s
연산지연	연산단위	부동소수점
	연산속도	10MB/s

3.2 기술적 해결 (Technical Solution)

기술적 해결 단계에서의 설계 결정은 다양한 제품 사양 중 최적의 제품 사양을 결정하는 활동과 제품 사양 결정 후 하드웨어/소프트웨어의 Co-Design에 대한 결정 활동이 이슈가 된다. 제품 사양의 경우 비기능적 요구사항에 따라 그 선택이 달라질 수 있다. 표 3은 표 2에 근거한 제품 사양 결정 과정을 보여주고 있다.

표 3 제품 사양 결정

구성 요소명	선택 대안 제품군	합계	근거 (만족도 100점 기준)					
			시스템대역폭		연산지원		성능	가격
			FSB대역폭	메모리대역폭	연산단위	연산속도		
마이크로	Intel8080	250	50	0	50	50	10	90
	Intel80386	370	100	50	50	50	50	70
	Intel80387	500	100	50	100	100	80	70
프로세스	Intel80486	550	100	100	100	100	100	50
	AMD386	290	50	50	50	10	30	100
	AMD486	550	100	100	100	80	90	80

표 3의 경우 대안 후보로 3개의 제품군으로 범위가 좁혀졌으며 추가적인 비기능적 요구사항 정보에 의해 최적의 제품 사양이 결정된다.

위와 같이 하드웨어 제품 사양이 비기능적 요구사항을 근거로 결정되면 하드웨어/소프트웨어 Co-Design에 대한 결정이 이루어진다. Co-Design 과정에서 하드웨어와 소프트웨어의 개발 범위가 결정된다. 개발 범위 결정 시에도 비기능적 요구사항이 사용된다. 임베디드 시스템은 하드웨어와 소프트웨어와 병렬적 개발과 함께 개발 중간에 제품 통합 과정이 점진적으로 이루어진다.

그림 4는 비기능적 요구사항을 근거로 한 하드웨어/소프트웨어 개발과 제품 통합을 보여주고 있다.

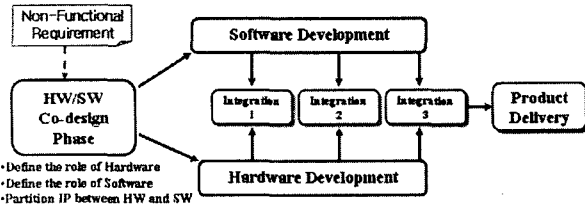


그림 4 HW/SW 개발 및 통합

3.3 제품 통합 (Product Integration)

제품 통합 단계에서의 설계 결정은 통합 순서를 정의하는 통합 절차 결정과 통합된 제품을 테스트하는 테스트 케이스 결정이 중요한 이슈가 된다.

하드웨어/소프트웨어 Co-Design 계획을 입력으로 하여 통합 절차 대안들이 작성된다. 이 대안 중 최적의 통합 절차를 결정한다. 제품 통합 절차 결정 시에는 기능적 요구사항 중 품질에 관계된 속성들이 중요한 근거가 된다. 표 4은 비기능적 요구사항을 근거로 통합 절차의 순서 결정 템플릿이다.

표 4 제품 통합 절차 순서 결정

통합 순서	통합환경	구성요소		근거	
		HW	SW	근거1
1					
...					

표 5는 표 4의 템플릿을 바탕으로 작성된 임베디드

시스템의 통합 절차 결정 예의 일부분이다.

표 4 제품 통합 절차 순서 결정

통합 순서	통합환경	구성요소			근거 (만족도 100점 기준)			
		HW	SW	합계	가용성	안정성	결합도	
1	dongle	타겟보드	부트로더	300	100	100	100	
1	오시로스 코프	마이크로 프로세스	RTOS	300	100	100	100	
1	JTAG	롬	RTOS	300	100	100	100	
1	JFlash	Flash메모리	RTOS	290	100	100	90	
1	JTAG	SDRAM	입출력 미들웨어	290	100	100	90	
2	Qplus	무선 이더넷	네트워크 미들웨어	220	80	70	70	
3	Qplus	터치스크린	GUI 미들웨어	180	70	60	50	
...								

테스트 케이스는 테스트에 대한 초점 결정 시에 비기능적 요구사항이 사용된다. 예를 들면 실시간성을 위해서는 다수의 입력을 입력하여 그 반응시간을 체크하는 테스트 케이스에 초점을 맞추고, 안정성의 경우에는 넓은 범위의 입력 값에 대한 결과 값을 체크하는 테스트 케이스에 초점을 맞추는 설계 결정을 하게 된다.

4. 결론 및 향후 연구

본 논문에서는 임베디드 시스템의 특징을 고려하여 임베디드 시스템의 비기능적 요구사항을 분석하여 이를 개발 단계별 설계 결정에 대한 근거로 활용하는 방안을 제시한다. 비기능적 요구사항을 각 단계별 설계 결정에 대한 근거로 사용함으로써 개발 단계별 설계 결정에 대한 추적이 가능하게 된다.

향후 연구로는 임베디드 시스템의 특성을 고려한 비기능적 요구사항을 정형적 방법을 통해 모델링하는 방법에 대한 연구와 모델에 대한 세부적인 활용방안이 제시되어야 할 것이다.

참고문헌

- [1] Qing Li, Caroline Yao, *Real-Time Concepts for Embedded Systems*, CMP Books, 2003
- [2] Arnold S. Berger, *Embedded System design*, CMP Books, 2002
- [3] O. Sokolsky, "Resource Modeling for Embedded System Design", Proc. IEEE WSTFEUS2004, pp.99~103, Vienna, Austria, May, 2004.
- [4] 김문희, 김두현, 김정국, "임베디드 시스템과 소프트웨어 공학", 정보과학회지, 제22권 제6호, pp.5~13, 2004.6.
- [5] Gerald Kotonya, Ian Sommerville, *Requirement Engineering*, WILEY, 1998.
- [6] Mary Beth Chrissis, Mike Konrad, Sandy Shrum, *CMMI : Guidelines for Process Integration and Product Improvement*, Addison-Wesley, 2003.
- [7] B. P. Douglass, *Real Time UML : Advances in the UML for Real-Time Systems 3rd Edition*, Addison-Wesley, 2004.
- [8] Dennis M. Ahern, Aaron Clouse, Richard Turner, *CMMI Distilled : A Practical Introduction to Integrated Process Improvement*, Addison-Wesley, 2003.