

임베디드 소프트웨어의 기민한 속성 주도 설계(Agile Attribute-Driven Design) 적용을 위한 통합 분석 기법

안민찬

고려대학교 컴퓨터학과

e-mail : novice10@korea.ac.kr

Integrated Analysis Method for applying for The Agile Attribute-Driven Design of Embedded Software

Minchan An

Dept. of Computer Science & Engineering,
Korea University

요 약

최근 다양한 분야에서 임베디드 소프트웨어의 비중이 증대함에 따라 품질 요구 사항이 증대하였지만, 현실적인 개발 일정은 점점 줄어들고 있다. 이에 본 논문에서는 임베디드 소프트웨어의 기민한 속성 주도 설계(Agile Attribute-Driven Design)를 체계적으로 적용하기 위한 ‘통합 분석 기법’을 정의한다. 그리고 이를 통한 기민한 설계 접근법을 제안한다. ‘통합 분석 기법’은 임베디드 소프트웨어의 특성을 고려한 기능 분석과 동시에 ‘품질 속성 시나리오’를 분석할 수 있는 기법으로서 고품질의 아키텍처 구축을 목적으로 한다. 또한 본 논문에서는 개미 로봇 구현 사례를 통해 ‘통합 분석 기법’을 검증하고 효과를 확인한다.

1. 서론

오늘날 임베디드 소프트웨어는 가전제품, 자동차, 핸드폰등 다양한 분야에서 그 비중이 증대되고 있다. 자연스럽게 그에 따르는 소프트웨어의 품질 요구사항도 같이 증가하고 있다. 이러한 품질 요구사항을 만족시키기 위해서는 임베디드 소프트웨어의 아키텍처 품질이 우선적으로 보장될 수 있어야 한다.

그러나 어플리케이션 소프트웨어와 달리 임베디드 소프트웨어는 하드웨어에 종속적이며, 실시간 반응과 고성능이 요구되는 특성상 공학적 접근으로 고품질 아키텍처를 개발하기에는 많은 노력이 필요하다[1]. 또한 임베디드 시스템의 짧아지는 개발 주기에 대응하기 위해서라도 기민한(agile) 고품질의 아키텍처를 개발할 수 있는 방법이 요구된다.

이에 본 논문에서는 임베디드 소프트웨어의 기민한 속성 주도 설계(AADD: Agile Attribute-Driven Design) 적용을 위해 ‘통합 분석 기법’을 제안한다. 1 장에서는

‘통합 분석 기법’의 필요성을 인식하고, 2 장에서는 전통적인 속성 주도 설계(ADD: Attribute-Driven Design) 접근법을 확인한다. 3 장에서 임베디드 소프트웨어의 AADD 적용을 위한 통합 분석 절차와 기법을 제안하고, 기민한 설계를 위한 연계성을 논의한다. 4 장에서는 ‘통합 분석 기법’을 개미 로봇 구현에 적용하고 검증한다. 5 장에서는 결론 및 향후 연구 방향을 논의한다.

2. 관련 연구

여러 가지 아키텍처 중심의 분석과 설계 방법론들은 10 년이 넘게 개발되고 적용되고 있다[2]. 그 중에서 ADD 는 품질 속성을 기반으로 한 아키텍처 설계 기법이다, 이를 위해서는 요구사항 분석과 더불어 품질 특성 시나리오 주도의 분석을 위해 품질 속성 워크샵(QAW: Quality Attribute Workshop)이 선행되어야 한다[3].

2.1 품질 속성 워크샵(QAW)

QAW 는 중요한 품질 속성과 시나리오를 식별하고, 우선 순위화하여, 소프트웨어 품질 요구사항을 정제하는 방법을 제공한다. 그림 1 은 QAW 수행을 위한 입출력 및 참석자들을 나타낸다[4].

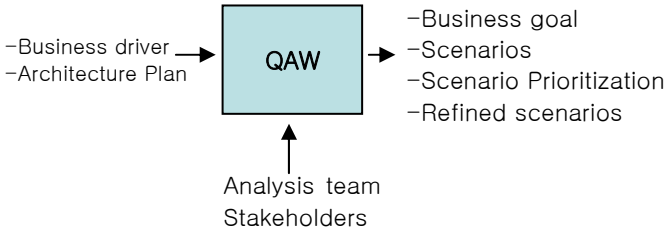


그림 1 QAW 입출력 및 참석자

QAW 에서 정의되는 ‘품질 속성 시나리오(Quality Attribute Scenario)’는 일반 시나리오와는 달리 특정한 시스템을 고려한 구체적인 품질 특성 요구사항이다. 품질 속성 시나리오는 표 1 과 같이 구성된다[3]

표 1 품질 속성 시나리오 구성

구분	설명
자극 소스	자극을 유발하는 실체(Entity)
자극 (Stimulus)	시스템이 고려해야 할 상태(Condition)
환경 (Environment)	자극이 발생시 시스템의 상태
대상물 (Artifact)	자극을 받는 대상
반응 (Response)	자극이 발생 후의 반응
반응 측정	반응에 대한 측정

2.2 속성 주도 설계(ADD)

ADD 는 재귀 분해(Recursive Decomposition) 절차를 통해 소프트웨어의 품질 속성을 만족시키는 소프트웨어 아키텍처를 정의하는 설계 방법이다. 이러한 재귀 분해 과정을 통해 품질 속성 시나리오들을 만족시키는 전술들(Tactics)과 아키텍처 패턴들을 적용한다. 그리고 이러한 패턴에 의해서 정의된 모듈 타입에 기능을 부여한다[3]. 그림 2 는 ADD 입출력 및 참석자를 보여준다[4].

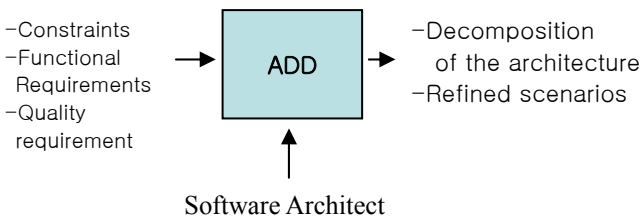


그림 2.ADD 입출력 및 참석자

3. 통합 분석 기법과 AADD 접근

‘통합 분석 기법’은 임베디드 소프트웨어의 특성을 고려한 기능 분석과 동시에 품질 속성 시나리오를 분석할 수 있는 기법이다. 그리고 이를 통해 AADD 접근법을 논한다.

3.1 통합 분석 절차

소프트웨어 분석을 위해서는 여러 가지 접근법이 있다[5]. 통합 분석 절차는 고전적인 방법이지만 임베디드 시스템의 특성을 고려한 기민한 개발을 위해서 구조적 분석 절차를 기반으로 수행된다[6]. 통합 분석 절차는 그림 3 과 같이 요구사항과 품질 속성 분석 절차가 병렬적으로 수행되면서 상호작용한다.

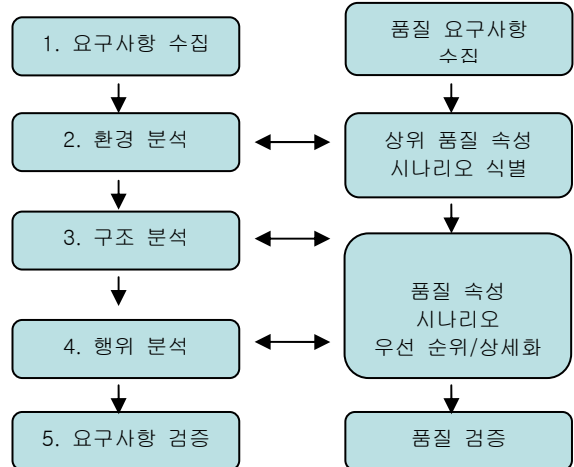


그림 3 통합 분석 절차

먼저 통합 분석 절차에서는 요구사항 수집과 동시에 품질 요구 수집이 수행되어야 한다. 그러나 개발 초기에는 요구사항뿐만 아니라, 품질 요구사항이 불명확할 수도 있다. 이러한 문제는 통합 분석 절차를 수행하면서 상세화 된다.

수집된 기능 및 품질 요구사항을 바탕으로 환경 분석을 수행하게 된다. 이때 환경 분석과 동시에 상위 품질 속성 시나리오가 식별된다. 그리고 식별된 상위 품질 속성 시나리오를 기반으로 구조 분석 및 행위 분석을 수행하게 된다. 이때 각각의 모듈들은 하향식 재귀 분해를 통해 품질 속성을 반영한 상세 분석이 이루어지게 된다. 또한 품질 속성 시나리오는 상세 분석 활동에 의해서 우선 순위화되고 상세화된다. 그리고 마지막으로 기능 및 품질 요구사항을 검증한다. 이때 잘못 수행된 활동들은 다시 처음으로 돌아가서 위의 절차들을 반복하게 된다.

3.2 통합 분석 기법

‘통합 분석 기법’은 환경 분석, 구조 분석, 행위 분석의 3 단계로 적용되는 기법이다.

먼저 환경 분석 단계에서는 개발하는 소프트웨어와 외부 실체(entity)와의 입출력 정보를 확인하기 위해 ‘이벤트 테이블’과 ‘시스템 내용도’가 정의된다[6]. 그리고 이와 동시에 상위 ‘품질 속성 시나리오’가 정의된다. 이러한 ‘품질 속성 시나리오’는 표 1 과 같이 구성된다.

구조 분석 단계에서는 환경 분석 단계의 결과물을 바탕으로 하향식 재귀 분해를 수행한다. 이때 사용되는 기법으로는 기능의 흐름과 제어 흐름을 보여주는 ‘데이터 흐름도’, ‘제어 흐름도’, 그리고 각각의 명세를 보여주는 ‘프로세스 및 제어 명세’가 사용된다[6]. 이와 동시에 ‘품질 속성 시나리오’는 우선 순위화되고 상세화된다.

마지막으로 행위 분석 단계에서는 구조 분석 단계의 결과물을 바탕으로 제어가 요구되는 부분을 중심으로 ‘상태 천이도’와 ‘상태 천이표’를 정의한다[6]. 구조 분석 단계와 마찬가지로 이때 ‘품질 속성 시나리오’는 우선 순위화되고 상세화된다. 단계별 통합 분석 기법들은 표 2와 같이 정리된다.

표 2 통합 분석 기법

단계	통합 분석 기법	
	요구사항 분석	품질 분석
환경 분석	-시스템 내용도 (System Context Diagram) -이벤트 테이블 (Event Table)	상위 품질 속성 시나리오 식별
구조 분석	-데이터 흐름도 (Data Flow Diagram) -제어 흐름도 (Control Flow Diagram) -프로세스/제어 명세 (PSPEC/CSPEC)	품질 속성 시나리오 우선 순위화 및 상세화
행위 분석	-상태 천이도 (State Transition Diagram) -상태 천이표 (State Transition Table)	품질 시나리오 우선 순위화 및 상세화

3.3 기민한 속성 주도 설계(AADD) 접근

AADD는 아키텍처 모델링을 효과적인 수행하기 위한 ADD 기반의 기민한 접근법이다. 프로젝트에 부담을 주지 않으면서 기민한 모델링을 효과적으로 수행하기 위해서는 단순성, 의사 소통, 피드백, 용기, 겸손의 가치가 요구된다.[7].

2장에서 보듯이 ADD를 적용하기 위해서는 요구사항 분석을 완료 후 QAW와 같은 품질 속성 시나리오 분석이 수행되어야 한다. 이는 리소스와 시간이 필요한 활동이다. 그러나 3장에서 살펴본 ‘통합 분석 기법’을 적용하면 기능 분석과 동시에 품질 속성 시나리오 분석이 수행되므로, 설계 단계에서 단순하지만 효과적으로 ADD를 적용할 수 있다. 또한 ‘통합 분석 기법’은 ADD와 같이 하향식 재귀 접근법을 사용하고 있으므로, ADD로 전환 시 개발자간 명확한 의사 소통과 피드백을 가능하게 한다. 또한 이를 통해 ADD 잘못된 접근법과 결과물을 쉽게 확인할 수 있게 하여 설계 접근법에 용기를 주고, 특정 품질 속성에 얽매이지 않고 겸손하게 ADD를 적용할 수 있게 한다.

즉 ‘통합 분석 기법’은 ADD 적용 시 단순하면서도 정확한 기민한 모델링을 가능하게 해준다. 이는 곧 고

품질의 아키텍처 개발을 가능하게 한다.

4. 통합 분석 기법 구현

6족 개미 로봇 시스템을 통해서 ‘통합 분석 기법’을 적용한다. 개미 로봇은 8비트 마이크로 컴퓨터에 의해서 제어되는 임베디드 시스템이다.

4.1 개미 로봇 요구사항[8,9]

개미 로봇은 눈 역할을 하는 한 개의 적외선 센서와 오른쪽과 왼쪽의 하나씩의 더듬이(antenna)를 가지고 있다. 그리고 3개의 모터에 6개의 발이 연결되어 전진, 후진, 왼쪽, 오른쪽으로 이동이 가능하다.

개미 로봇은 이동 중에 적외선 센서나 더듬이 센서에 의해서 장애물이 인식되면 미리 정해진 프로그램에 따라서 개미 로봇은 회전하거나 후진하면서 장애물을 피할 수 있어야 한다. 그림 4는 개미 로봇 시스템을 보여준다

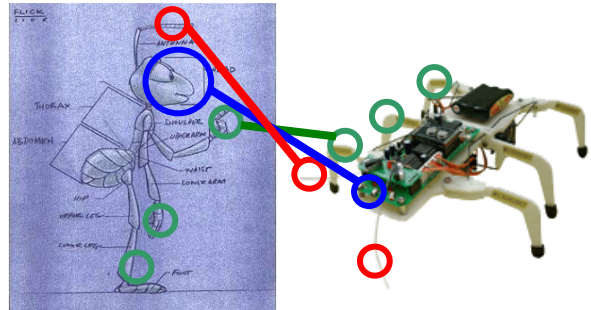


그림 4. 개미 로봇 시스템

4.2 통합 분석 기법 적용

먼저 개미 로봇의 요구사항을 바탕으로 환경 분석을 수행할 수 있다. 그 결과인 ‘시스템 내용도’는 그림 5와 같이 정리된다[9].

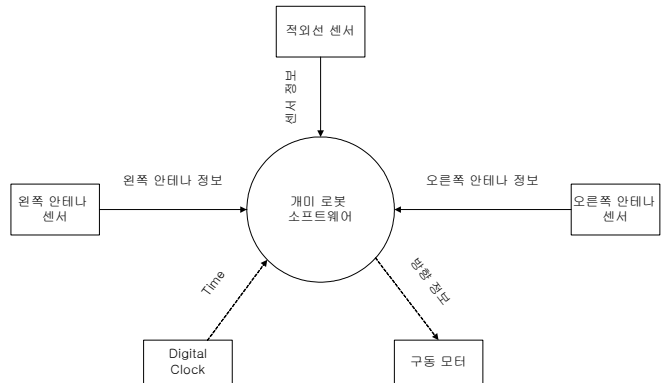


그림 5 개미 로봇 시스템 내용도

그리고 구조 분석 결과를 바탕으로 표 3과 같이 상위 품질 속성 시나리오를 식별할 수 있다. 개미 로봇의 여러 가지 품질 속성들이 존재하겠지만, 가용성(Availability), 성능(Performance), 변경성(Modifiability)에 대한 일부분만을 정의하였다.

표 3 상위 품질 속성 시나리오

	품질 속성 식별		
	가용성	성능	변경성
소스	적외선 센서	적외선 센서	개발자
자극	적외선 센서 동작 에러	적외선 센서 식별 정보	적외선 센서 변경
환경	작동 시	작동 시	개발 시
대상물	장애물 인식 모듈	제어 모듈	장애물 인식 모듈
반응	제어 모듈에 센서 상태 정보를 제공 함.	3 초 안에 장애물을 피한다	적외선 센서 변경시 다른 모듈에 영향 이 없음.
반응 측정	적외선 센서 정보 제공율	작동 시간	센서 변경 영향도

위의 환경 분석 결과와 상위 품질 속성 시나리오를 바탕으로 개미 로봇의 구조와 행위 분석이 수행된다. 이와 동시에 상위 품질 속성 시나리오는 구체화되고 우선 순위화 된다. 따라서 개미 로봇의 장애물 식별 모듈에는 품질 속성인 가용성을 만족시키기 위해 적외선 센서 동작 상태를 확인할 수 있는 기능이 추가된다. 이러한 내용은 그림 6 과 같이 분석된다. 그림 6 은 개미 로봇의 여러 가지 구조 분석 결과 중 장애물을 식별하는 모듈을 보여준다.

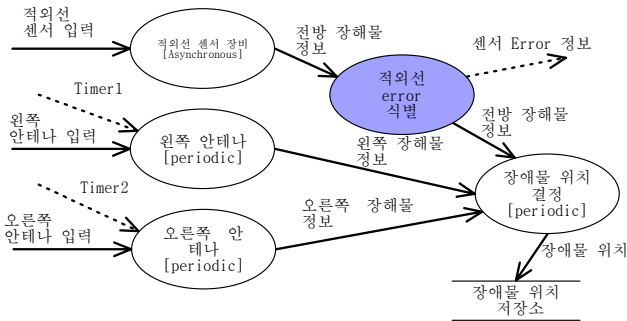


그림 6 개미 로봇 장애물 식별 모듈

또한 개미 로봇 제어 모듈에서도 가용성을 만족시키기 위해서 새로운 상태가 더 요구된다. 즉 전진, 후진, 좌회전, 우회전의 본래 기능의 상태 이외에도 센서 에러 복구를 위한 상태가 더 필요하다. 그림 7 은 행위 분석 결과 중 로봇 제어 모듈의 '상태 천이도'를 보여준다.

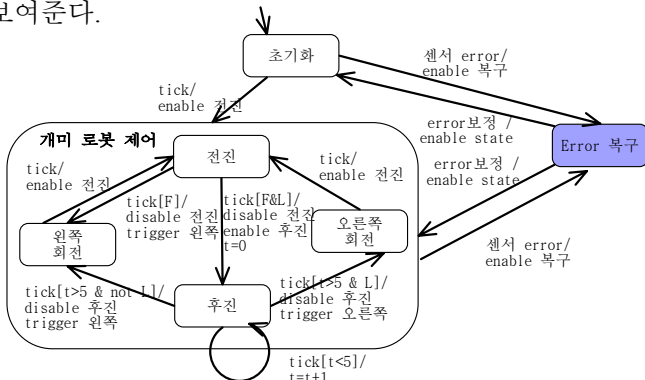


그림 7 개미 로봇 제어 모듈

5. 결론 및 향후 연구 방향

본 논문에서는 임베디드 소프트웨어에 AADD 를 적용을 위해 분석 단계에서부터 품질 속성 시나리오를 고려한 통합 분석 절차 및 기법을 논의하고, 개미 로봇 시스템을 통해 ‘통합 분석 기법’을 검증하였다.

하드웨어에 종속적인 임베디드 소프트웨어의 특성 상 품질 속성을 분석 단계에서 동시에 분석하지 않으면, 분석 단계 후 나중에 다른 활동으로 설계 품질 속성을 식별하고 설계 활동에 반영하는 것은 어려운 일이다. 이에 본 논문에서 제시한 ‘통합 분석 기법’을 활용하면 효과적인 접근이 용이하며, 임베디드 시스템의 실질적인 기민한 설계 모델링이 가능하게 된다.

향후 연구 과제로는 품질 속성 시나리오를 고려한 통합 분석 기법의 패턴에 대한 연구와 이를 통한 기민한 모델링과의 구체적인 연계 기법에 대한 연구가 필요하다.

참고 문헌

- [1] Bruce Powel Douglass, ‘Real Time UML’, 3rd Ed. Addison Wesley, 2004
- [2] Rick Kazman, Robert L. Nord, Mark Klein, ‘A Life Cycle View of Architecture Analysis and Design Method’, CMU/SEI-2003-TN-026, September 2003
- [3] Len Bass, Paul Clements, Rick Kazman ‘Software Architecture in Practice’, 2nd Ed, Addison-Wesley, 2003
- [4] Robert L. Nord, William G Wood, Paul C Clements, ‘Integrating the Quality Attribute Workshop (QAW) and the Attribute-Driven Design (ADD) Method’. CMU/SEI-2004- TN-017, July 2004
- [5] Lilianan Dobrica, Eila Niemela, ‘A Survey on Software Architecture Analysis Methods’, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL 28, NO.7, JULY 2002
- [6] Hassan Gomaa, ‘Software Design Methods for Concurrent and Real-time Systems’. Addison-Wesley, 1996
- [7] Scott Ambler, ‘Agile Modeling: Effective Practices for eXtreme Programming and the unified process, John Wiley & Sons, 2002
- [8] 신대섭, 정상봉, ‘초보자가 만드는 로봇트’, 도서출판세화, 2002
- [9] 민상운, Structured Analysis/Structured Design: Real-time 세미나, Solution Link, 2002 4 월