



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사 학위논문

소프트웨어 기능안전성 확보를 위한 위험분석과
소프트웨어 요구분석 및 항해장비 사례 연구

Risk Analysis and Software Requirement Analysis for
Software Functional Safety and a Case Study on
Echo Sounder

지도교수 이 서 정

2017년 2월

한국해양대학교 대학원

컴퓨터공학과

임상우

본 논문을 임상우의 공학석사 학위논문으로 인준함.



위원장 박 휴 찬 (인)

위원 이 영 찬 (인)

위원 이 서 정 (인)

2016년 12월 28일

한국해양대학교 대학원

목 차

List of Tables	iv
List of Figures	v
Abstract	vi

1. 서 론

1.1 연구 배경 및 필요성	1
1.2 연구 목적	3

2. 관련 연구

2.1 IEC 61508의 소프트웨어 기능안전성	4
2.2 소프트웨어 기능안전성 사고 사례	7
2.3 IMO MSC Circ 1512와 실무적용지침	14
2.3.1 IMO MSC Circ 1512	14
2.3.2 해양 SQA/HCD 실무적용지침	17
2.4 소프트웨어 안전성 공통 개발 가이드	19
2.5 HAZOP 프로세스	26

3. 기능안전성 확보를 위한 위험분석과 소프트웨어 요구분석	
3.1 위험분석	30
3.1.1 위험분석의 고려사항	30
3.1.2 추출 기초조사	31
3.1.3 요구사항 추출	32
3.1.4 요구사항 평가	34
3.2 소프트웨어 요구분석	37
3.2.1 소프트웨어 요구분석의 고려사항	37
3.2.2 이해관계자 인식	37
3.2.3 요구사항 도출	38
3.2.4 요구사항 상세내역 작성	38
4. 사례 연구	
4.1 선박 시스템 음향 측심기(Echo sounder)	42
4.2 음향 측심기의 위험분석 절차 적용	46
4.2.1 추출 기초조사	46
4.2.2 요구사항 추출	48
4.2.3 요구사항 평가	50
4.3 음향 측심기의 소프트웨어 요구분석 절차 적용	53
4.3.1 이해관계자 인식	53
4.3.2 요구사항 도출	53
4.3.3 요구사항 상세내역 작성	53
5. 결론 및 향후과제	58
참고문헌	59

List of Tables

Table 2.1	Configuration of IEC 61508	5
Table 2.2	List of IMO e-Navigation SQA/HCD guideline	18
Table 2.3	Configuration of Safety common development guideline	19
Table 2.4	Output list of V-model process	23
Table 2.5	Verified analysis technique of Software responsibility · safety	24
Table 2.6	Tool list of Software responsibility · safety	25
Table 3.1	Comparative analysis table by process	29
Table 3.2	Example template of Guideword	32
Table 3.3	Example template of HAZOP worksheet	33
Table 3.4	Degree of Hazard alert(Marine vessel accident)	34
Table 3.5	Example template of Hazard analysis result	36
Table 3.6	List of Requirements related Software	38
Table 3.7	Item list of Requirement specification	39
Table 3.8	Template of Requirement specification	41
Table 4.1	Performance standard of Echo sounder	43
Table 4.2	Software category of Echo sounder function	46
Table 4.3	Guideword List of Echo sounder	47
Table 4.4	Guideword of Echo sounder	48
Table 4.5	HAZOP worksheet of Echo sounder	49
Table 4.6	Degree of Hazard alert(Maritime equipment software)	50
Table 4.7	Safety-conscious HAZOP worksheet of Echo sounder	52
Table 4.8	Template of Requirement specification of Echo sounder(Main body)	54
Table 4.9	Template of Requirement specification of Echo sounder(Transducer)	57

List of Figures

Fig. 1.1 Functional Safety of E/E/PE safety-related system	1
Fig. 2.1 Accident investigation of Therac 25	7
Fig. 2.2 The report of Therac 25 at the time of the accident	8
Fig. 2.3 Software accident investigation of Therac 25	9
Fig. 2.4 Error message of Therac 25	10
Fig. 2.5 The photograph of Korean Air 801 crash	11
Fig. 2.6 The photograph of Washington subway accident in 2009	13
Fig. 2.7 Concepts and standards for e-Navigation quality design attributes	15
Fig. 2.8 Overview of software quality activities for e-Navigation system	16
Fig. 2.9 Process of V-model	22
Fig. 2.10 Process of HAZOP	27
Fig. 3.1 Diagram of requirement extraction activity	31
Fig. 4.1 Photograph of Echo sounder	42

Risk Analysis and Software Requirement Analysis for Software Functional Safety and a Case Study on Maritime Equipment

Lim, Sang Woo

Department of Computer Engineering
Graduate School of Korea Maritime and Ocean University

Abstract

With an increase in the proportion of software in systems used in each industry, software safety-related incidents are increasing. To solve this issue, major industrial sectors such as railways, aerospace and medical services are working to secure software safety in accordance with IEC 61508-based functional safety standard. However, since the standard document only presents a list of outputs and declarative sentences, it is difficult to directly apply it practical business affairs, In addition, hardware-centered functional safety standards are difficult to apply to software. The common development guide for software safety published by the National IT Industry Promotion Agency in 2016 provides common guidelines that can be utilized in various fields based on the IEC 61508 standard, but additional customizing is required to apply to a specific field or scale. In this regard, this paper attempted to define processes that can be applied to small-scale software and outputs of each stage and to apply the software safety process to an echo sounder, which is a device for marine communications, as an example of the maritime sector.

KEY WORDS: Software Development Process, Software Development Life Cycle, Software Quality Assurance, Functional Safety, Echo Sounder



제 1 장 서론

1.1 연구 배경 및 필요성

IEC 61508:2010 (Functional Safety of Electrical / Electronic / Programmable Electronic safety-related systems)에서는 기능안전성을 소프트웨어의 기능에 위험이 생기거나 사고가 날 염려로부터의 자유로 정의하고 있다(IEC 61508, 2010). 다양한 분야에서 대부분의 장치가 소프트웨어를 내장함에 따라 소프트웨어의 안전성에 대한 중요도가 높아지고 있다. 해외 선진국에서는 2000년 초반부터 이를 인식하고 원전, 항공, 의료, 철도, 장치산업 등의 다양한 분야에서 IEC 61508을 기반으로 하는 안전 표준들을 작성하여 기능안전, 신뢰성, 품질 및 성능에 대한 검증을 요구하고 있으며 점차 모든 분야로 확산하고 있다. Fig. 1.1은 IEC 61508을 기반으로 하는 각 분야의 표준들을 나타낸다.

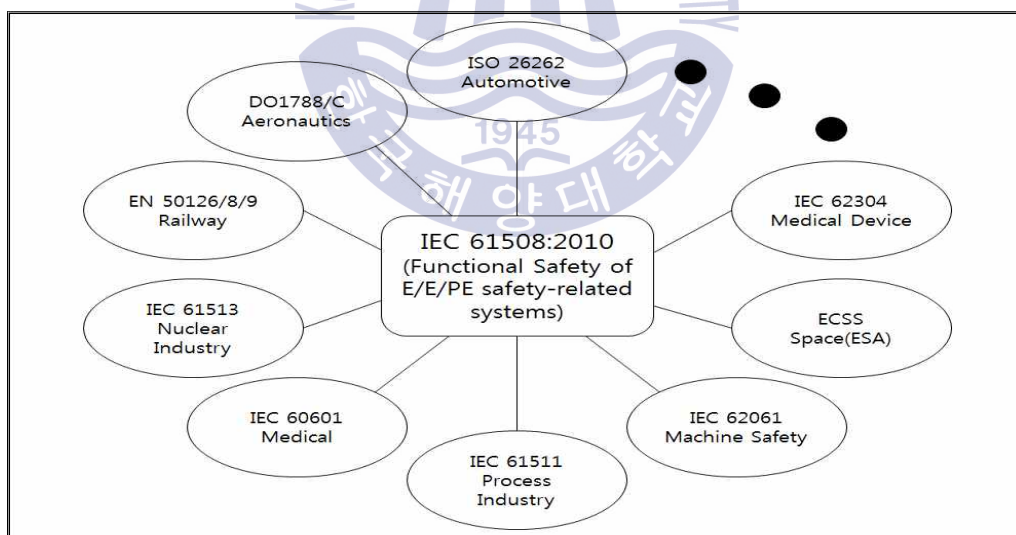


Fig. 1.1 Functional Safety of E/E/PE safety-related system

- (1) ISO 26262 (Road vehicles - Functional safety): 자동차 기능안전성 국제 표준으로 자동차에 탑재되는 전자장비 시스템의 오류로 인한 사고방지를 위해 ISO에서 제정한 자동차 기능안전 국제 규격
- (2) DO 178B (Software Considerations in Airborne System and Equipment Certification): 항공기 시스템과 장비 인증에 관한 소프트웨어 고려사항으로 RTCA사에 의해 발표된 소프트웨어 개발 표준
- (3) EN 50126/8/9 (Railway Applications): 신뢰성, 가용성, 유지보수 및 안전에 대한 명세서
- (4) IEC 61513 (Nuclear Power Plants): 시스템의 일반적인 요구사항의 안전에 중요한 계측 및 제어에 관한 표준
- (5) IEC 60601 (Medical Electrical Equipment): 안전 및 의료 전기 장비의 효율성에 대한 기술 표준
- (6) IEC 61511 (Functional Safety): 안전성을 보장하기 위한 시스템 공학 방법을 설명하는 기술 표준
- (7) IEC 62061 (Safety of Machinery): E/E/PE 제어 시스템의 기능안전 표준
- (8) ECSS (European Cooperation for Space Standardization): 유럽의 우주분야를 개선하기 위한 표준
- (9) IEC 62304 (Medical Device Software): 의료장치 내에서 의료 소프트웨어 및 소프트웨어 생애주기의 요구사항을 규정하는 표준

선박·해양 분야는 다른 분야와 달리 IEC 61508을 기반으로 하는 안전표준이 없으므로, 조선·해양 분야 기술 표준화는 ISO(International Organization for Standardization: 국제 표준화 기구)에서 관리하는 표준규격과 IMO(International Maritime Organization)에서 결의되는 강제적 규정만을 포함하여 이루어지고 있다. 미래창조부 산하 NIPA (National IT Industry Promotion Agency: 정보산업진흥원)에서는 국제 안전 표준이 지정되지 않은 조선 분야에 IEC 61508을 기반으

로 하는 ‘소프트웨어 안전성 공통 가이드’를 개발하여 산업 현장에 적용하도록 하고 있다.

1.2 연구 목적

본 논문은 모든 종류의 산업에 적용 가능한 기본적인 기능안전 표준인 IEC 61508의 최신 버전을 분석하여 아직 표준이 정의되지 않은 조선·해양 분야에도 적용할 수 있도록 하는 것을 목적으로 한다.

본 논문에서는 기존의 소프트웨어 개발 생애주기에서 위험분석을 위한 단계를 추가하여 소프트웨어의 기능안전성을 만족할 수 있도록 한다. 위험분석 단계는 소프트웨어의 위험을 사전에 분석하는 단계이다. 소프트웨어 요구분석 단계는 기존의 활동을 진행함과 동시에 위험분석 단계에서 분석한 결과를 기반으로 위험요인을 도출하는 모델을 수립하고, 그에 맞는 템플릿을 정의한다. 모델의 검증을 위해서는 항해장비인 음향 측심기(Echo Sounder)에 대한 사례연구를 실시하여 실무자들의 이해를 도울 수 있도록 한다. 사례 연구에서는 음향 측심기의 위험분석 요소와 요구사항 명세를 도출하고, 요구사항 명세서에 대한 템플릿을 작성하여 소프트웨어 공통 안전성 개발 가이드라인에 기반이 될 수 있도록 한다.

본 논문의 구성은 다음과 같다. 2장에서는 IEC 61508의 소개, 소프트웨어의 안전성 분야의 필요성을 증명하기 위한 안전성 관련 사고사례, 기존의 해양 소프트웨어의 개발 프로세스, 소프트웨어 안전성 공통 개발 가이드라인 소개, HAZOP(HAZard and OPerability analysis) 프로세스의 소개 및 절차를 설명한다. 3장에서는 관련 연구들을 적용하여 위험분석과 소프트웨어 요구분석의 절차 및 활동을 설명하고, 도출되는 산출물의 템플릿을 제시한다. 4장에서는 해양장비인 음향 측심기를 대상으로 위험분석을 시행, 산출물을 작성하였다. 요구사항 명세서에는 위험도를 추가함으로써 기능안전성을 만족시킬 수 있도록 하였다.

제 2 장 관련 연구

본 장에서는 조선·해양 분야에서 소프트웨어 기능안전성의 필요성을 보여주기 위한 사고 사례와 관련 연구들을 설명한다. 사고사례와 포함된 그림은 NIPA의 소프트웨어 신뢰·안전성 확보를 위한 공통 지침 및 가이드 개발 사업에서 발췌하였다(NIPA, 2016).

2.1 IEC 61508의 소프트웨어 기능안전성

IEC 61508은 모든 종류의 산업에 적용하기 위한 국제표준으로 명칭은 E/E/PE 전자 안전 관리 시스템의 기능안전이다. IEC 61508에서는 기능안전을 다음과 같이 정의한다.

“E/E/PE 전자 안전 관련 시스템의 정확한 기능, 다른 기술 안전 관련 시스템과 외부적인 위험 감소 설비에 의존하는 제어 대상 장비와 제어 대상 장비를 제어하는 시스템 관련된 부분적 또는 전반적인 안전”

IEC 61508은 Table 2.1과 같이 안전생명주기, 하드웨어, 소프트웨어 등 3가지에 대한 안전성 구현방법 및 검증 방법을 제시하고 있으며, 소프트웨어 관련 부분은 Part 3에서 다루고 있다. 안전관련 시스템은 IEC 61508에서 정의된 안전수명주기에 따라 위험분석 및 평가, 안전무결성수준 (SIL: Safety Integrity Level)을 설정하고, 하드웨어와 소프트웨어를 목표로 한 수준에 충족할 수 있도록 구현하며, 설치, 운영, 유지보수, 변경, 폐기까지 관리한다.

Table 2.1 Configuration of IEC 61508

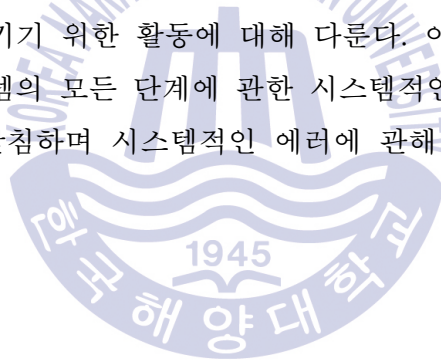
구분	구성
Part 0	기능안전성과 IEC 61508
Part 1	일반 요구사항
Part 2	E/E/PE 가능한 전자장치 안전 관련 시스템의 요구사항
Part 3	소프트웨어 요구사항
Part 4	정의와 약어
Part 5	안전무결성수준 결정 방법의 예
Part 6	IEC 61508의 Part 2와 Part 3의 적용 지침
Part 7	기법과 수단의 개요

IEC 61508의 각 파트에 대한 설명은 다음과 같다.

- Part 0 (기능안전성과 IEC 61508): 기능안전에 대한 일반적인 개념을 정의하고 각 Part의 개요 및 구성에 관하여 명시
- Part 1 (일반 요구사항): 안전 수명주기에 따른 목적, 적용 범위, 입력 및 산출물을 명시함으로써 기능안전을 위한 전체 프레임워크를 정의
- Part 2 (E/E/PE 전자장치 안전 관련 시스템의 요구사항): 안전제어시스템에 요구되는 안전 요구사항을 결정하기 위한 다양한 기법의 적용 및 안전무결성수준의 등급화에 관한 내용
- Part 3 (소프트웨어 요구사항): 안전제어시스템을 개발하면서 하드웨어가 아닌 소프트웨어에 적용되는 안전기능과 안전무결성수준에 대하여 명시
- Part 4 (정의와 약어): IEC 61508에서 사용되는 용어에 대한 정의 및 설명
- Part 5 (안전무결성수준 결정 방법의 예): 리스크와 안전무결성의 개념에 대한 설명과 함께 두 개념 간의 관계를 명시하고, 안전무결성을 결정할 수 있는 다양한 방법론들에 관하여 예시를 들어서 설명

- Part 6 (IEC 61508의 Part 2, Part 3의 적용 지침): Part 2와 Part 3 적용의 기능적 단계와 두 가지 작동 모드에서의 하드웨어 고장 확률 평가 기법, 진단 범위 및 안전 고장비율 계산 예시 및 소프트웨어 안전무결성에 관하여 명시
- Part 7 (기법과 수단의 개요): Part 2 및 Part 3과 관련된 다양한 안전기법에 관한 설명을 제공함으로써 하드웨어 우발 고장에 대한 제어, 시스템 고장의 회피, 소프트웨어 안전무결성 달성을 위한 기법 및 방법 등을 명시

IEC 61508에서는 안전수명주기와 이에 따른 활동, 절차, 기술을 정의하고 있다. 위험검증과 안전무결성수준을 만족하는 설계를 위해 위험요인 분석 시행, 위험감소 대상을 식별하고 ISO 9001과 같이 조직, 프로세스, 인적자격요소 등이 식별된 위험을 감소시키기 위한 활동에 대해 다룬다. 이 표준에서는 안전 수명 주기의 사용으로 시스템의 모든 단계에 관한 시스템적인 상태에 적용되는 안전을 보증하는 것을 뒷받침하며 시스템적인 에러에 관해 가능성을 감소시키려는 목적을 갖는다.



2.2 소프트웨어 기능안전성 사고 사례

안전이 필수적으로 요구되는 시스템의 소프트웨어 신뢰·안전성 문제에 의한 사고들이 많이 존재한다. 이 사고들의 공통점은 한 번의 사고로 대규모의 재산 피해를 낼 수 있다는 점이다. 최근 소프트웨어 비중의 증가에 비해, 소프트웨어 복잡성으로 인해 신뢰·안전성을 확보하기는 어렵다. 다음은 소프트웨어의 신뢰·안전성에 의한 사고 사례들이다.

(1) Therac 25 사고 사례

Fig 2.1은 Therac 25 사고에 대한 사진이다.

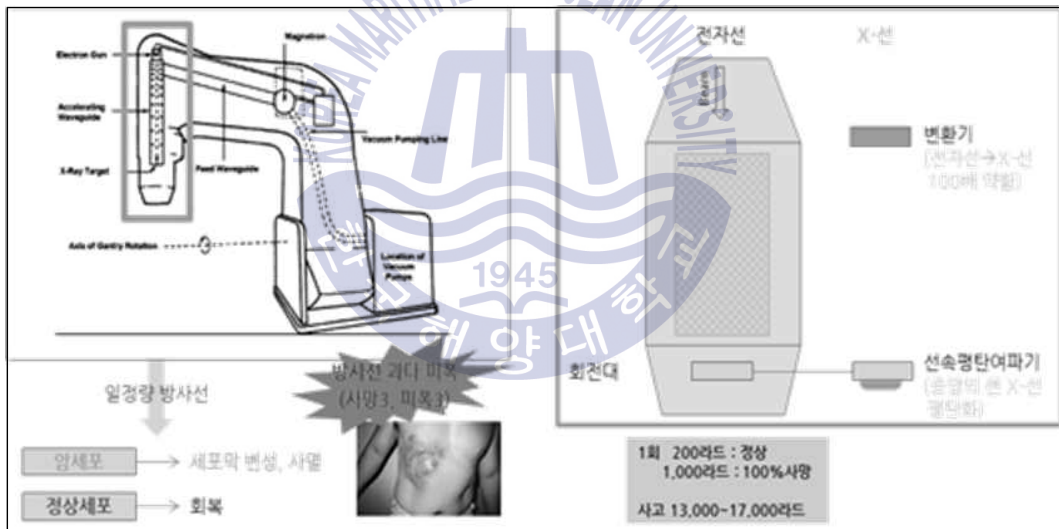


Fig. 2.1 Accident investigation of Therac 25

Therac 25는 피부 근처의 종양을 제거하는 Electron 모드와 피부 깊숙한 곳의 종양을 제거하는 X-ray 모드를 통합한 혁신적 제품이었다(Nancy Leveson, 1995). 하지만, “안전기능 미비”로 3명이 사망하고, 3명은 심각한 방사능 후유증에 시달리게 되었다. Fig.2.2은 사고 당시의 보고서이며, 사고 발생 경위는 다

음과 같다.

PATIENT NAME:TEST	BEAM TYPE: X ENERGY(KeV):	A	1
TREATMENT MODE: FIX		25	
	ACTUAL	PERSCRIBED	
UNIT RATE/MINUTE	0	200	
MONITOR UNITS	50 50	200	
TIME (MIN)	0.27	1.00	
GANTRY ROTATION (DEG)	0.0	0	VERIFIED
COLLIMATOR ROTATION (DEG)	359.2	359	VERIFIED
COLLIMATOR X(CM)	14.2	143	VERIFIED
COLLIMATOR Y(CM)	27.2	273	VERIFIED
WEDGE NUMBER	1	1	VERIFIED
ACCESSORY NUMBER	0	0	VERIFIED
DATE: 84 OCT-26	SYSTEM: BEAM READY	OP.MODE: TREAT	AUTO
TIME: 12:55. 8	TREAT: TREAT PAUSE	X-RAY	173777
OPR ID: T25V02-R03	REASON: OPERATOR	COMMAND:	

Fig. 2.2 The report of Therac 25 at the time of the accident

- 환자의 치료를 위해 처방 값을 X선 모드로 입력
- 전자선 모드로 변경 후에 자동으로 나머지 값을 그대로 사용
- “방사선 조사 준비됨” 을 확인 후 그대로 치료 시행
- 방사선 조사 후 정지
- 오류 메시지 출력 (오류 메시지 해석 불가)
- 치료 중지, 사용자의 재입력
- 다시 치료 수행

Therac 25 사고는 기기에 대한 정보 부족, 사용자의 실수, 해석 불가능한 오류 표시 등으로 인한 문제들로 인해 사고가 발생하였다. Fig.2.3은 Therac 25의 분석 결과이며, 정리된 사고의 원인은 다음과 같다.

① 소프트웨어

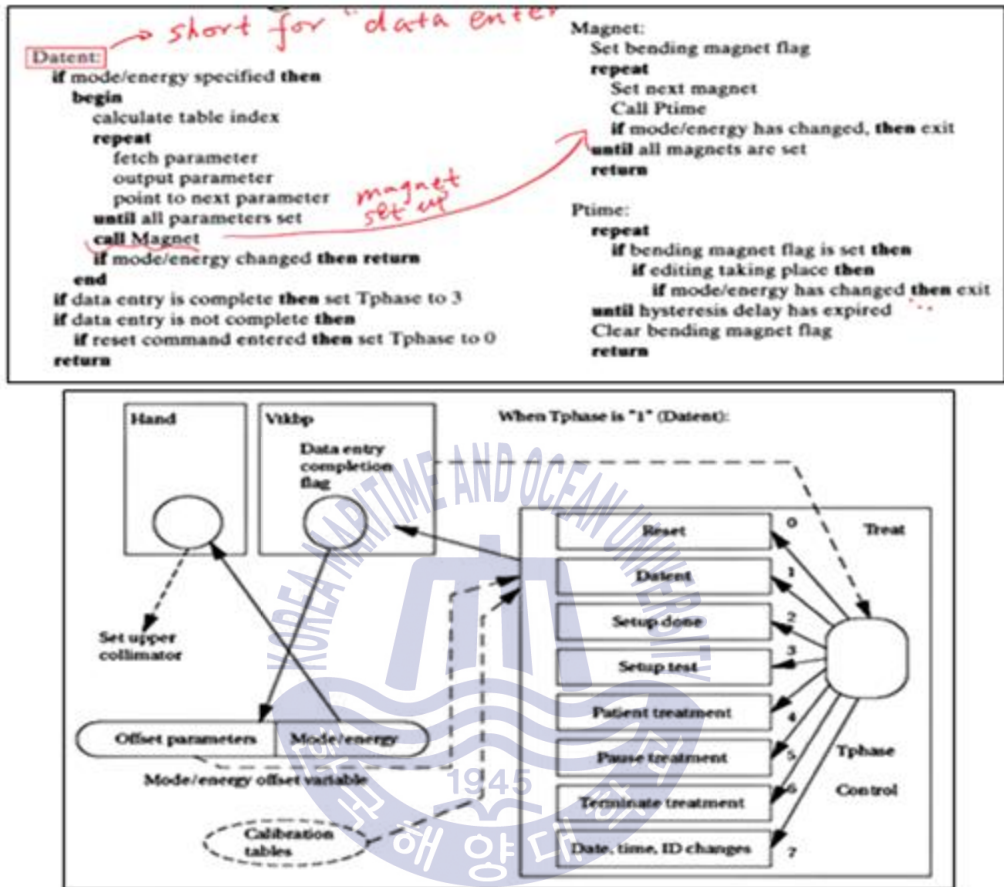


Fig. 2.3 Software accident investigation of Therac 25

방사선 치료기는 선형가속기를 이용하고 선형가속기는 전자를 가속해 만든 전자선으로 방사선 치료를 진행하는 방식이다. 전자선 치료 시에는 산란박을 사용하고 X선 치료 시에는 ‘선속평탄여과기’ 라는 장치를 사용하게 되는데, Therac 25는 이 두 장치와 방사선이 조사할 위치를 확인하는 장치인 ‘필드라이트’ 를 사용하였다. Therac 25 이전 모델은 하나뿐인 전자선 출구를 회전판 위에 위치시켜 하드웨어를 통해 회전판을 돌려가며 필요한 장치를 사용하였지만, Therac 25는 소프트웨어가 제어하였고 소프트웨어의 오류로 인해 턴테이블의

위치를 제대로 제어하지 못하여 사고가 발생하였다.

② 사용자 인터페이스

Therac 25는 기기실에서 기기를 수동으로 조정하는 구조였다. 사용자가 제어실에서 처방 값을 모두 입력하는 과정을 개선하기 위해 DEFAULT 기능을 추가하여 이전 값을 자동으로 입력하는 편의 기능을 추가하게 되었다. 하지만 이러한 구성으로 인해 방사선사들은 소프트웨어의 오류 메시지의 출력을 늦게 발견하였으며 오류 메시지 (MALFUNCTION 54)를 해석하지 못한 상태로 기기를 계속 구동하였다.

③ 문서의 구성

Fig. 2.4는 Therac 25의 설명서에 나열된 오류 메시지이다.

Jan. 17	Second overdose at axlma
Jan. 26	AECL sends FD, their revised test plan
Feb.	Hamilton clinic investigates first accident and concludes there was an overdose.
Feb. 3	AECL announces changes to Therac-25
Feb. 10	FDA sends notice of Adverse Findings to AECL declaring Therac-25 defective under U.S. law and asking AECL to notify customers that machine should not be used for routine therapy. Health Protection Branch of Canada does the same thing. This lasts until August 1987.
Mar. 5	AECL sends third revision of CAP to FDA
Mar.	Second user's group meeting
Apr. 9	FDA responds to CAP and asks for additional information
May 1	AECL sends fourth revision of CAP to FDA
May 26	FDA approves CAP subject to final testing and safety analysis.
June 5	AECL sends final test plan and draft safety analysis to FDA
July	Third user's group meeting
July 21	Fifth (and final) revision of CAP sent to FDA

Fig. 2.4 Error message of Therac 25

사고 장비에 제공되는 기기-운영자 설명서에는 오류의 종류만 나열되고 메시지에 대한 설명이 나타나 있지 않았다. 오류 메시지 중에서는 환자에게 위험할 수 있다는 메시지도 포함만 되어있을 뿐 따로 분류되지 않은 상태였기 때문에 방사선사들은 “MALFUNCTION 54”라는 메시지를 무시하게 되었다.

④ 제조 기업(AECL)의 무지

사고 발생 후 제조 기업에서는 소프트웨어에 대한 지나친 과신으로 검사를 제대로 하지 않았다. 당시 기술로는 시스템에 문제가 발생하였을 경우 하드웨어만 검사하여 원인을 찾기 힘들었고 코드에 대한 3자 리뷰가 없어 소프트웨어의 원인 탐색이 힘든 상황이었다. 결국, 소프트웨어의 신뢰성 및 안전성 개념의 무지로 인해 사고가 발생하였다.

(2) 대한항공 801편 추락사고 사례

Fig. 2.5는 대한항공 801편 추락사고 현장 사진이다.



Fig. 2.5 The photograph of Korean Air 801 crash

대한항공 801편 추락 사고는 1997년 8월 6일 대한민국 김포국제공항에서 출발한 801편이 미국의 괌에 있는 아가나 국제공항에서 착륙에 실패, 추락하여 승객 237명과 승무원 17명을 합쳐 총 254명 중 228명이 사망하였고, 총 26명이 다친 사고이다(Albert G. Reitan, 1998). 사고 발생 경위는 다음과 같다.

비행기에 활공각 지시기의 신호가 잡혀 기장과 부기장의 혼란

→ 고도 확인 절차를 생략함과 동시에 규정 고도를 무시

→ 801편은 계속 하강하면 활주로가 보일 것이라 판단, 고도를 순식간에 하강했고, 날씨가 좋지 않아 고도를 내려도 활주로가 보이지 않은 것이라 판단

→ 충돌 전 대지접근 경보장치에서 경보가 여러 번 울렸고, 부기장도 “접근 실패”를 외쳤지만, 기장은 경보와 부기장이 외치는 말을 무시

→ 기장은 “고 어라운드”라고 말하며 복행 선언을 했지만, 관성의 법칙으로 인해 메인 랜딩기어가 송유관을 친 뒤 뒷바퀴부터 추락 후 화재 발생

사고의 주원인은 조종사의 실수이지만, 만약 소프트웨어가 비행기가 안전고도보다 낮게 비행하고 있음을 인지하고 경보를 울렸다면, 참사를 막을 수 있던 사고이다. 정리된 사고의 원인은 다음과 같다.

① 괄 국제공항의 최저 안전고도 경고 시스템의 결함

괄 국제공항의 최저 안전고도 경고 시스템은 괄의 레이더를 중심으로 반경 100~101.8km 사이의 불과 1.8km 영역에서만 정상 동작하는 시스템이다. 대한항공이 신호를 수신한 영역은 괄에서 멀리 떨어진 바다로, 항공기가 낮게 비행할 이유가 없는 영역이므로 관여되지 않았다. 당시 괄의 레이더 반경 100km 안에 있던 대한항공 801편이 안전고도보다 낮게 비행했음에도 경보가 발생하지 않는 결함이 발생하였다.

② 아우터 마커의 신호 무시

아우터 마커는 상공을 향해 수직으로 전파를 발사하는 장치로, 조종사는 아우터 마커의 신호 수신 및 고도를 확인한다. 하지만 기장은 착륙시도를 하면서 아우터 마커의 신호를 확인하지 않아 규정 고도보다 낮게 날고 있는 것을 인지하지 못하였다.

③ 글라이드 슬롭(항공기의 안전착륙을 돕는 전자장치)의 허위 신호

팜 공항의 글라이드 슬롭 장치가 고장 난 상태임을 승무원 모두가 인지하고 있었지만 수리되지 않은 상태로 글라이드 슬롭이 전파 간섭에 의한 허위신호를 송출하게 되었다. 활주로 접근 중에 글라이드 슬롭의 신호가 수신되었지만 누구도 해당 신호의 사실 여부를 관제탑에 문의하지 않았으며 글라이드 슬롭 고장 시 유지되어야 하는 고도 규정을 무시하고 계속 하강을 시도하여 사고가 발생하였다.

(3) 2009년 워싱턴 지하철 탈선사고 사례

Fig. 2.6은 2009년 워싱턴 지하철 탈선사고 현장 사진이다.



Fig. 2.6 The photograph of Washington subway accident in 2009

2009년 6월 22일 워싱턴과 메릴랜드를 연결하는 지상구간에 포트토티역에 진입하기 위해 선로에 정차했던 214호 열차를 뒤따르던 112호 열차가 들이받은 사고이다(NTSB, 2009). 사고로 112호 열차가 214열차 위로 올라타면서 탈선하여 9명이 사망하고 70명이 중경상을 입었다. 사고 경위는 다음과 같다.

구형 모델인 사고 열차의 1,2번째 제동장치의 정기점검 미준수(교체비용이 너무 많이 든다는 이유로 일부 제동장치와 비상탈출구만 개선하고 제동장치를 교체하지 않아 사고 발생)

→ 수동조작 중이던 214호는 저속도로 운행 중이었으며, 주변 건축물 때문에 112호 열차에서 214호를 감지하지 못함

→ 소프트웨어로 구현한 자동운행 모드에서 소프트웨어 오류로 뒤따라오던 열차를 정지시키지 못하고, 비상브레이크도 작동하지 않음

사람이 육안으로 인지하지 못하는 예상 사고 시점에 도움을 주기 위해 소프트웨어가 추가되었지만, 관리 미숙과 결함을 발견하지 못해 일어난 사고이다.

2.3 IMO MSC Circ 1512와 실무적용지침

2.3.1 IMO MSC Circ 1512

IMO에서 e-Navigation 전략의 도입으로 새로운 소프트웨어와 시스템이 많이 개발되고 있으며 여러 장비 간의 통합과 여러 정보가 전송되고 표현되기 때문에 소프트웨어의 중요성이 강조되고 있다. 사용자들에게 좋은 품질의 소프트웨어와 사용하기 편한 시스템을 제공하기 위해 e-Navigation 소프트웨어 품질 보증에 대한 제안이 이루어졌으며 2015년 6월 MSC(Maritime Safety Committee) 95차 회의에서 SQA(Software Quality Assurance)/HCD(Human Centred Design) 통합 가이드라인을 회람문서로 최종 승인하였다(IMO, 2015). 이 가이드라인을 위한 실무지침(practical guidance)은 ISO/IEC 12207의 소프트웨어 개발 프로세스와 ISO/IEC 15288의 시스템 개발 프로세스를 따라서 작성되었다(Lee, *et al.*, 2015).

이 가이드라인은 설계 시 고려되어야 하는 품질 특성으로 제품 및 데이터 품질, 사용자 요구 충족, 보안, 기능안전성을 포함한다. Fig. 2.7은 e-Navigation 시스템의 설계자 및 개발자가 모든 품질 속성을 해결하고 전체 시스템 품질을 보장하기 위해 고려해야 하는 관련 표준들을 나타낸다.

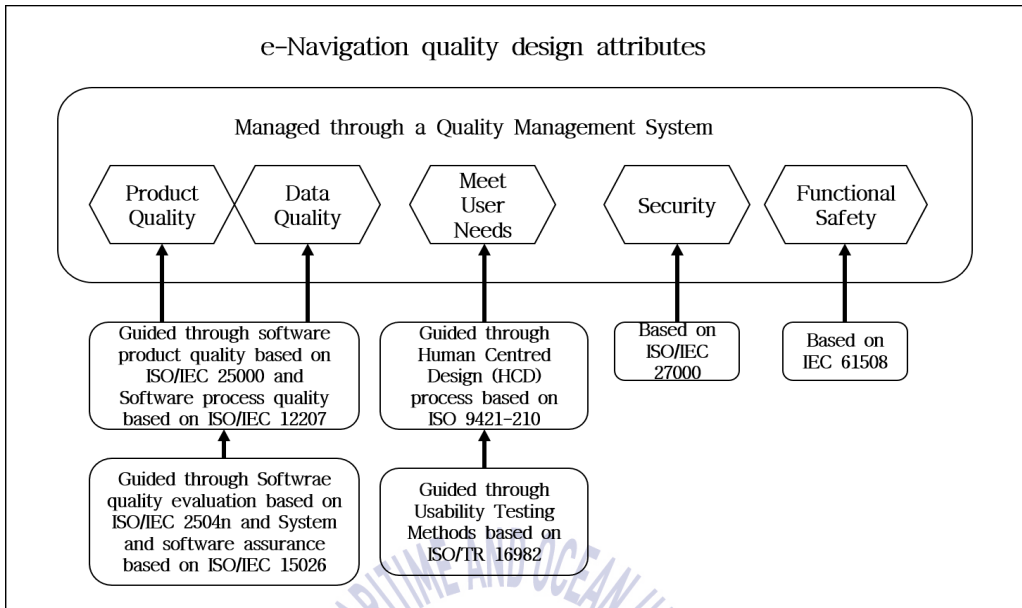


Fig. 2.7 Concepts and standards for e-Navigation quality design attributes

Fig. 2.8은 2014년 3월에 IMO NCSR(Navigation, Communications and Search and Rescue)회의에서 제시 및 승인된 소프트웨어 품질을 위한 e-Navigation SQA/HCD 가이드라인의 소프트웨어 품질을 위한 활동을 나타낸다(IMO, 2015).

각 단계에서 수행하는 활동은 다음과 같다.

(1) Pre-activity: 예비 위험분석 시행

- 소프트웨어를 개발, 운용할 때에 생길 수 있는 위험원을 사전에 분석하는 활동이다.

(2) Activity 1: 이해관계자 및 시스템 요구사항 정의

- 이 활동은 요구되는 특성을 구체화하고 개발 중인 시스템의 사용 맥락을 확인하는 것을 포함한다. 이 활동 중에 시스템의 검증 및 적합성 요구사항이 확인되어야 한다.

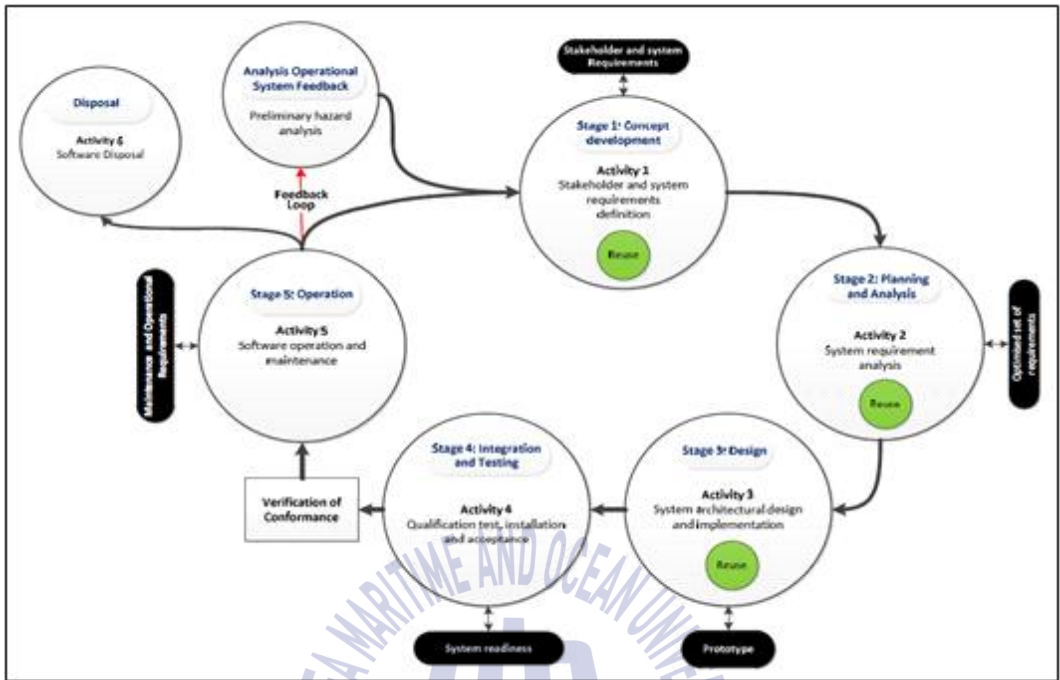


Fig. 2.8 Overview of Software Quality activities for e-Navigation system

(3) Activity 2: 시스템 요구사항 분석

- 이 활동은 최적의 솔루션을 보장하기 위해 개발된 다양한 구성을 가진 기능적, 비기능적 시스템 요구사항을 정의한다. 이 활동의 결과로 시스템 요구사항의 우선순위 선정, 승인, 업데이트된다.

(4) Activity 3: 소프트웨어 아키텍처 설계 및 구현

- 이 활동은 시스템의 요소를 정의하고 구성하며, 소프트웨어의 품질이 요구사항을 충족하는지를 보장하는 과정을 포함한다.

(5) Activity 4: 소프트웨어 테스트, 설치 및 적용지원

- 이 활동은 통합된 소프트웨어가 시스템 요구사항을 준수하는 지를 확인한다. 소프트웨어의 성능기준 검토의 신뢰성, 무결성을 보장하기 위해 적절한 소프트웨어 테스트 방법 및 기준이 개발되어야 한다.

(6) Activity 5: 소프트웨어 운용 및 유지보수

- 이 활동은 정해진 환경에서 소프트웨어가 올바르게 작동하는지를 식별하고 평가하는 과정을 포함한다. 운용 및 유지보수 전략은 소프트웨어 개발자와 사용자 간의 협의를 통해 개발될 필요가 있다.

(7) Activity 6: 시스템 폐기

- 시스템 폐기 전략은 지식 보존 및 장기적인 영향의 분석을 쉽게 하기 위해 개발되는 과정이다.

2.3.2 해양 SQA/HCD 실무적용지침

한국해양대학교에서는 IMO MSC Circ 1512에서 정의한 가이드라인을 기반으로 하여 e-Navigation SQA/HCD 통합 실무적용지침을 개발하여 산업체에 적용하기 위해 노력해왔다. 2015년 2월에는 통합 가이드라인 워크숍, 4월에는 국제 워크숍을 개최하여 가이드라인 개발에 참여했던 국제 전문가들과 조선 분야 관계자들로부터 산업계 적용에 관한 다양한 의견을 수집하였다(Kim, H., 2015). 그 결과로 2016년 5월에 실무적용지침이 개발되었다. Table 2.2는 실무적용지침이 제공하는 내용을 나타낸 것이다.

가이드라인과 실무적용지침은 SQA와 HCD를 위한 제품 및 데이터 품질, 사용자 요구 충족이외의 품질특성인 보안성과 기능안전성은 별도의 방법으로 품질을 만족시키는 것을 권고하고 있다. 2016년 12월에는 IEC 61508을 기반으로 하는 소프트웨어 안전성 공통 개발 가이드를 개발하여 기능안전성을 확보할 방법을 제시하였다.

Table 2.2 List of IMO e-Navigation SQA/HCD guideline

IMO e-Navigation SQA/HCD 실무적용지침	
Part 1	1. 소프트웨어 개발 방법론
	2. SQA의 세 가지 품질 모델
Part 2	액티비티 별 기대 산출물 목록 요약표
	1. 이해관계자 요구사항 분석 액티비티
	2. 시스템 요구사항 분석 액티비티
	3. 시스템 아키텍처 디자인 액티비티
	4. 구현 액티비티
	5. 시스템 통합 및 테스트 액티비티
	6. 시스템 자질 테스트 액티비티
	7. 소프트웨어 설치 액티비티
	8. 소프트웨어 적용 지원 액티비티
9. 소프트웨어 운용 액티비티	
Part 3	10. 소프트웨어 유지보수 액티비티
	11. 소프트웨어 재사용 액티비티
	12. 소프트웨어 폐기 액티비티
Appendix	A. 용어 및 정의
	B. Reference
	C. 관련 기관들의 소개 및 역할
	D. Action의 상세 설명과 Templates
	E. IMO 통합 가이드라인

2.4 소프트웨어 안전성 공통 개발 가이드

소프트웨어 분야는 실제 고장률 측정이 어려우므로 요구사항에 따른 활동을 정의하고 그 증거를 확보하여 이의 달성 정도인 안전무결성수준을 관리한다(한국방송통신전파진흥원, 2013). IEC 61508에서는 기능안전성 확보를 위한 방법을 다루었지만, 표준문서의 특성상 필요한 산출물만이 제시되어 있을 뿐, 실행방법에 대한 명세는 작성되어 있지 않다. 미래창조과학부 산하 NIPA에서는 기능안전성을 국내의 모든 분야의 소프트웨어에 도입하고자 소프트웨어 안전성 공통 개발 가이드라인을 개발하였다. 소프트웨어 안전성 공통 개발 가이드라인은 현장에서 활용하기 위해 개념 제시 수준을 넘어 실제 실무에 활용할 수 있도록 구체적인 방안을 사례 기반으로 작성되었다. 가이드라인은 국제 기능안전성 표준인 IEC 61508을 기반으로 작성되어 Table 2.3과 같이 구성된다.

Table 2.3 Configuration of Safety common development guideline

I. 서론	
제 1장	배경
제 2장	가이드 목적 및 적용 범위
제 3장	문서의 구성
제 4장	용어 정의
II. 기능안전성 이해	
제 1장	소프트웨어 기능안전성
제 2장	안전성 관리
III. 소프트웨어 기능안전 생애주기	
제 1장	개요
제 2장	개발 생애주기 구성
제 3장	안전한 소프트웨어 개발 기법
제 4장	안전한 소프트웨어 개발 양식

가이드라인에서 제안한 생애주기는 소프트웨어 개발 방법론으로 V모델 개발 방법론을 따른다. V모델은 소프트웨어 계획수립을 시작으로 운영하는 단계까지 총 7단계로 구성되어 있으므로 위험분석을 위한 추가활동이 다른 개발 방법론보다 쉽다. 각 단계의 수행활동은 개발활동, 검증 및 확인활동, 안전 활동들로 구성되어 있으며 각 단계에서 도출되는 결과는 산출물로서 관리되어야 한다. V모델의 각 단계에서 수행하는 활동은 다음과 같다.

(1) 소프트웨어 개발계획(Software Plan phase) / 위험분석(Risk Analysis phase): 소프트웨어의 개발계획을 수립하는 단계이며 안전 활동에서는 소프트웨어 안전 분석, 안전계획 수립의 활동을 진행한다.

(2) 소프트웨어 요구분석(Software Requirements specification phase): 소프트웨어의 요구사항을 명세하여 추적성을 유지하도록 하는 단계이며 안전 활동에서는 소프트웨어 요구사항의 안전평가를 수행하며 이에 따른 안전기록을 작성하는 활동을 진행한다.

(3) 소프트웨어 설계(Software Design phase): 소프트웨어의 구조를 설계하여 설계명세서를 작성하고, 테스트 명세서를 작성하는 단계이며 안전 활동에서는 소프트웨어 안전평가를 수행하며 모듈안전평가를 진행한다.

(4) 소프트웨어 구현(Software Construction phase): 소프트웨어의 모듈을 구현하는 단계이며 안전 활동으로는 소프트웨어 구현 안전평가를 수행하며 이에 따른 안전기록을 작성하는 활동을 진행한다.

(5) 소프트웨어 통합(Software Integration phase): 소프트웨어의 모듈테스트, 통합테스트를 개발 및 수행하는 단계이며 안전 활동으로는 소프트웨어 통합 안전평가를 수행하며 이에 따른 안전기록을 작성하는 활동을 진행한다.

(6) 검증 확인(Validation & Verification): 요구사항을 통한 소프트웨어의 기능을 대상으로 소프트웨어의 성능기준을 검토하는 단계이며 안전 활동으로는 소프트웨어의 안전 분석에 대한 평가를 수행하며 이에 따른 안전기록을 작성하는 활동을 진행한다.

(7) 소프트웨어 운영(Software Maintenance phase): 소프트웨어의 운영, 유지 보수 및 수리를 진행하는 단계이며 안전 활동으로는 소프트웨어 운영 안전평가를 수행하며 이에 따른 안전기록을 작성하는 활동을 진행한다.

소프트웨어의 기능안전성은 하드웨어보다 요구사항 분석 단계에서의 활동이 중요하다. 그 이유는 소프트웨어 설계 이후의 단계들은 안전성의 분석이 아닌 적용을 실시하는 단계이므로 안전성에 대한 고려를 추가할 수 없기 때문이다. 소프트웨어의 안전성을 위한 고려사항을 요구사항 단계에서 명시하기 위해서는 이전단계에서 미리 식별되고 정의되어야 한다. 따라서 소프트웨어 기능안전성에 대한 위험분석 및 위험도 선정은 요구사항 단계 이전에 시행되며 이를 위험분석 단계로 정의한다.

본 논문에서는 위험분석 및 위험도 선정을 위해 V-모델의 단계를 Fig. 2.9의 점선 박스 범위인 소프트웨어 위험분석(RA) 단계와 소프트웨어 요구분석(SR) 단계의 절차를 다룬다. 위험분석 단계는 소프트웨어가 가질 수 있는 위험원을 분석 및 정의하는 단계이며 이 단계에서는 소프트웨어 오류를 발생하게 하는 위험원을 미리 정의함으로써 요구사항에서 나타날 수 있는 오류와 안전조치를 식별할 수 있도록 하는 것을 목적으로 한다. 위험원을 식별한 후, 소프트웨어 요구분석 단계에서 도출된 위험원을 고려하여 요구사항 명세서를 작성한다. 요구사항 명세서에는 요구사항에 따른 위험도와 이를 대비하는 안전 조치를 명시함으로써 위험발생 시, 신속한 대응이 가능하다.

Table 2.4는 ‘소프트웨어 안전성 공동개발 가이드’에서 각 단계에서 도출되어야 할 산출물과 각 항목의 수를 보여준다. 총 7단계로 20개의 산출물을 작성하고, 각 산출물에 포함된 항목은 164개로 구성되어 있다. 각 산출물에 대한 식별번호는 단계를 진행하였을 때, 필수적으로 도출되어야 하며 식별번호는 프로젝트 초기에 산출물 명명규칙에 의해 정의된다.

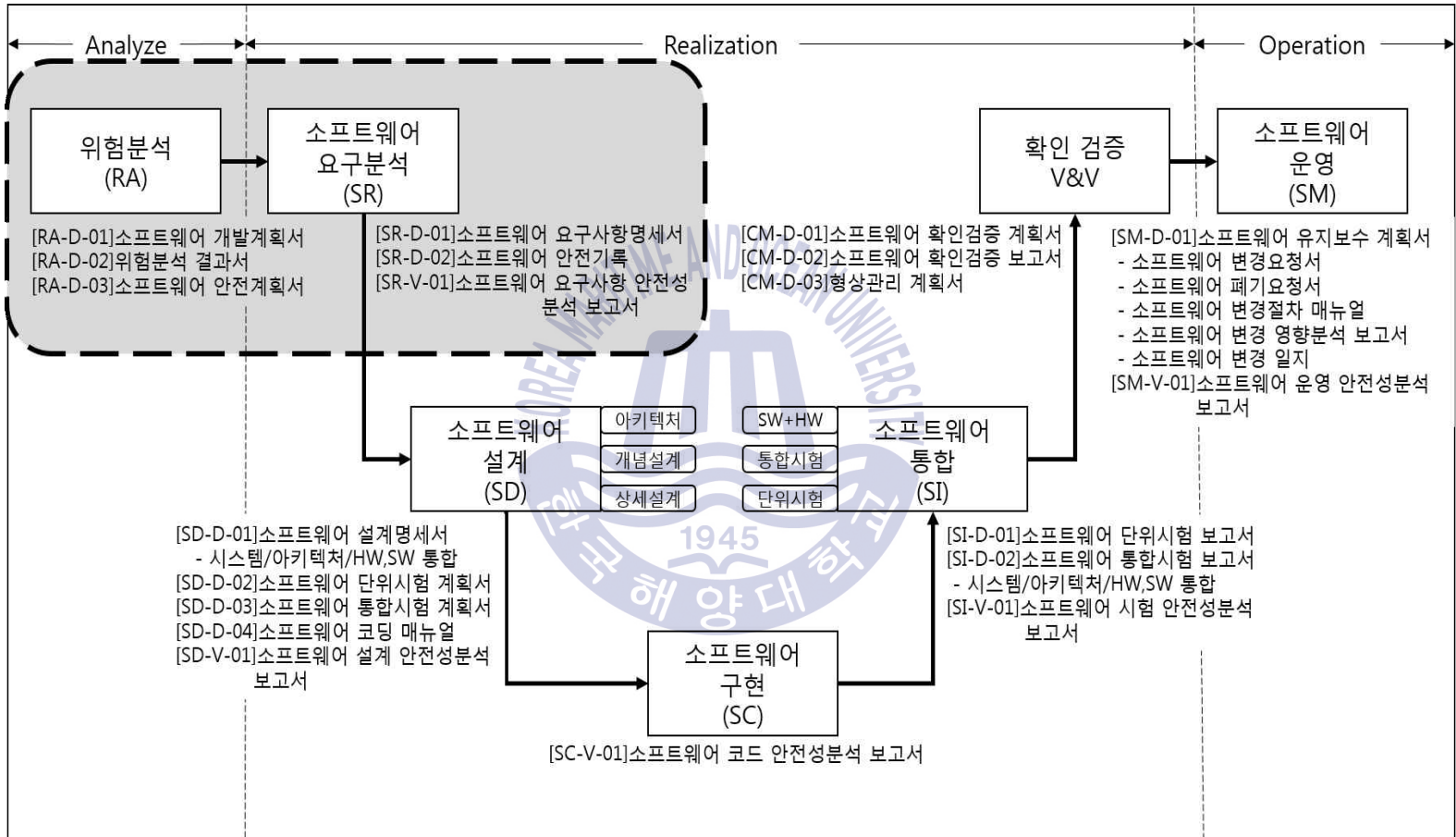


Fig. 2.9 Process of V-model

Table 2.4 Output list of V-model process

단계	식별번호	산출물	
		내역	항목
위험분석 (RA)	RA-D-01	소프트웨어 개발계획서	8
	RA-D-02	위험분석 결과서	
	RA-D-03	소프트웨어 안전계획서	6
소프트웨어 요구분석(SR)	SR-D-01	소프트웨어 요구사항명세서	12
	SR-D-02	소프트웨어 안전기록	7
	SR-V-01	소프트웨어 요구사항 안전성분석 보고서	9
소프트웨어 설계(SD)	SD-D-01	소프트웨어 설계명세서	6
	SD-D-02	소프트웨어 단위시험 계획서	12
	SD-D-03	소프트웨어 통합시험 계획서	13
	SD-D-04	소프트웨어 코딩 매뉴얼	7
	SD-V-01	소프트웨어 설계 안전성분석 보고서	9
소프트웨어 구현(SC)	SC-V-01	소프트웨어 코드 안전성분석 보고서	9
소프트웨어 통합(SI)	SI-D-01	소프트웨어 단위시험 보고서	9
	SI-D-02	소프트웨어 통합시험 보고서	9
	SI-V-01	소프트웨어 시험 안전성분석 보고서	9
검증 확인 (V&V)	CM-D-01	소프트웨어 확인검증 계획서	8
	CM-D-02	소프트웨어 확인검증 보고서	4
	CM-D-03	형상관리 계획서	9
소프트웨어 운영(SM)	SM-D-01	소프트웨어 유지보수 계획서	9
	SM-V-01	소프트웨어 운영 안전성분석 보고서	9

안전성 공통 개발 가이드에서는 소프트웨어의 신뢰·안전성 분석을 위해 검증된 분석기법을 사용하여 실무에서 활용 가능한 방안과 함께 구체적인 내용에 대해서는 관련 도구를 제시한다. Table 2.5는 소프트웨어 신뢰·안전성의 분석 기법 예시를 나타내며, Table 2.6은 소프트웨어 신뢰·안전성 점검 도구 사용 현황을 나타낸다(소프트웨어정책연구소, 2016).

Table 2.5 Verified analysis technique of Software responsibility · Safety

소프트웨어 테스트 기술	Logic-based 테스트
	Concolic 테스트
소프트웨어 안전성 분석 기법	Preliminary Hazard Analysis(PHA) & SoftWare Safety Assessment(SWSA)
	Fault Tree Analysis(FTA)
	Failure Mode and Effects Analysis(FMEA)
	HAZard and OPerability analysis(HAZOP)
	System Theoretic Process Analysis(STPA)

소프트웨어 테스트 기술에는 Logic-based 테스트 기법과 Concolic 테스트 기법이 있다. Logic-based 테스트는 테스트의 기준을 참/거짓으로 분류하여 실시한다. 각 단계의 참/거짓 값의 모든 조합의 경우를 테스트하기 때문에 확장성에서 큰 문제가 있어 실질적으로 거의 사용되지 않는다. Concolic 테스트는 소프트웨어의 분기와 반복을 고려하여 다양한 행동을 판단하여 테스트한다. 일반적인 Concolic 테스트 기법은 조건을 풀어낼 수 없는 경우 종료하지만 이를 해결하기 위한 다양한 하이브리드 기법도 등장하고 있다.

소프트웨어 안전성 분석 기법은 다음과 같이 분류한다(이장수, 2015). PHA 기법은 밝혀낸 위험과 원인 등을 제거하기 위한 목적으로 분석을 수행하며 분석의 결과는 정성적으로 표현하는 기법이다. 이와 유사한 SWSA 기법은 소프트웨어 개발의 초기 단계에서 작업서를 작성하여 분석을 수행하는 기법이다. FTA 기법은 분석하고자 하는 결과를 최상위에 놓고 그 원인을 하위로 연결하면서 수행하는 기법이다. FMEA 기법은 시스템의 특정 부분이 일으키는 고장 유형이 전체 시스템에 미치는 영향을 확인하는 방법으로 원인으로부터 결과를 도출하는 귀납적 추론 방법을 이용한 기법이다. HAZOP 기법은 시스템의 기능이나 요건과 같은 매개변수와 안내단어를 조합하여 예상하지 못한 위험을 도출하고 시스템의 안전에 미치는 영향을 분석하는 기법이다. STPA 기법은 STAMP(System Theoretic Accident Modeling and Process)를 기반으로 하는 기법으로 시스템 요소 간의 상호작용을 방해하는 부적절한 행위를 분석하는 기법이다. 논문에서는

HAZOP 기법을 선정하였으며 자세한 설명은 2.5절에서 다룬다.

Table 2.6 Tool list of Software responsibility · Safety

구분	내용
안전	<ul style="list-style-type: none"> · 안전 프로세스 관리, 안전성 분석을 위한 도구(일부 존재) · SIL 프로세스 관리 시스템 운영 도구
품질	<ul style="list-style-type: none"> · 소프트웨어 성능, 신뢰성에 대한 시험 항목과 측정도구 · 소프트웨어 자동화 테스트 도구 · 소프트웨어 요구사항/형상/변경/이슈/빌드/ 배포 관리 도구 · 소프트웨어 개발 생명주기 영역에 대한 테스트, 리스크 도구 · 프로젝트 매니지먼트 도구 · 소프트웨어 개발/테스트 주체 커뮤니케이션 도구: 버그 트래킹 시스템 도구 등

안전성 분석을 위한 도구와 함께 안전무결성등급에 따른 프로세스 관리 시스템 운영 도구를 사용하지만, 전문 분석 도구의 기반이 미약하다. 품질을 위한 도구는 품질 측면에서 다양한 도구가 활용되고 있지만, 테스트 자동화 도구들에 대한 수요가 높아 더 많은 도구의 등장이 필요하다.

‘소프트웨어 안전성 공통 개발 가이드’에서는 소규모 프로젝트에서 발생하기 쉬운 문제점을 고려하여 애자일방법론을 권고하고 있다. 이는 1) 불충분한 계획, 2) 낮은 우선순위, 3) 경험이 부족한 프로젝트팀, 4) 동시에 여러 가지 역할을 담당하는 프로젝트 매니저, 5) 소규모 프로젝트의 특성을 고려하지 않은 기존 방법론의 다양한 문제를 해결할 수 있다. 또한, 인간 중심적, 커뮤니케이션 지향적이며, 시장의 변화에 유연하게 대응할 수 있는 장점이 있다(소프트웨어정책연구소, 2016).

2.5 HAZOP 기법

위험을 분석하는 기법은 PHA&SWSA, FTA, FMEA, HAZOP 등 여러 가지가 존재한다. HAZOP 기법은 귀납적 추론 방식에 기반을 두어 워크시트 작성을 통해 분석을 수행하는 기법으로 위험식별에 더욱 유용한 기법이다. HAZOP 기법은 인력과 장비에 위험을 나타낼 수 있는 문제를 파악하고 평가를 목적으로 한다. HAZOP 기법을 사용할 때는 공정에 관련된 여러 분야의 전문가들이 모여서 공정에 관련된 자료를 토대로 정해진 방법에 의해서 원래 설계된 목적에 어긋나게 된 원인과 결과를 식별하고 위험에 야기되는 문제 가능성의 유무를 파악하는 것이 중요하다.

HAZOP 기법은 시스템이 설계된 대로 작동하는 동안 근본적인 위험성이나 운용상의 문제가 없는 경우를 대상으로 시작한다. 위험원이 발생하였을 때 시스템이나 소프트웨어가 이를 처리하는 과정에서 사고가 발생하는지를 논의한다. HAZOP 기법은 회의를 통해 결과로 나타나기 때문에 정성적인 결과 값이 도출된다. 따라서 정확한 결과를 위해서는 정량적인 값의 적용이 필수적이다. Fig 2.10은 HAZOP 프로세스를 나타낸 것이며 프로세스별 수행내용은 3장에서 다룬다.

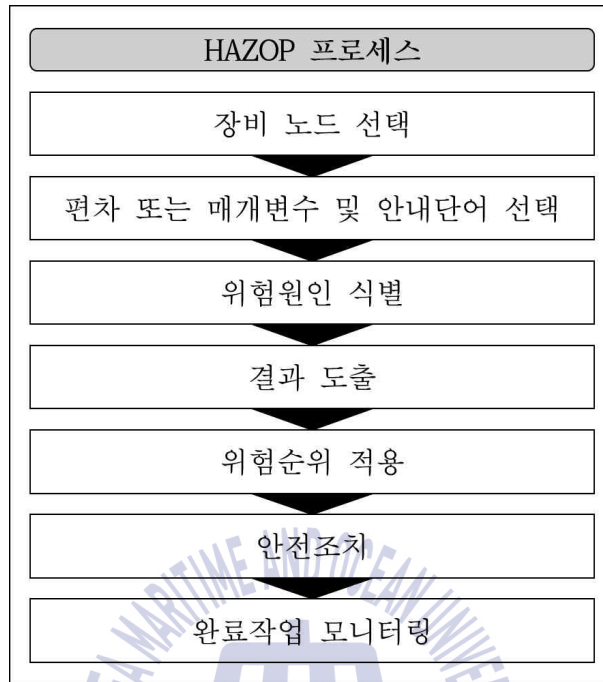
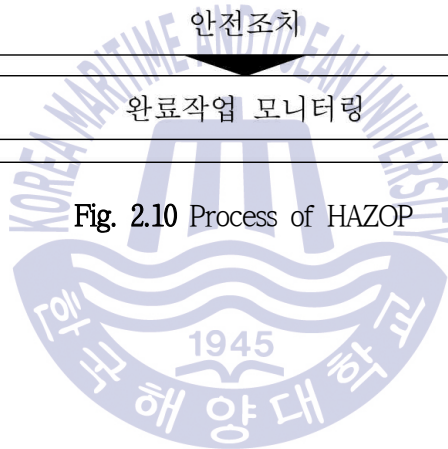


Fig. 2.10 Process of HAZOP



제 3 장 기능안전성 확보를 위한 위험분석과 소프트웨어 요구분석

기존의 소프트웨어 생애주기에서는 기능안전성 확보를 위해 위험분석의 중요성은 인지하고 있지만 수행절차의 내용과 산출물의 도출 방식이 명확하지는 않다. NIPA에서는 기능안전성 확보를 위한 절차와 산출물을 명시하고 국내의 모든 산업 분야에서 적용할 수 있는 소프트웨어 안전성 공통 개발 가이드라인을 제시하였다. 가이드라인은 Table 2.4와 같이 총 7단계로 20개의 산출물을 작성하고, 각 산출물에 포함된 항목은 164개로 구성되어 있어 모든 규모의 소프트웨어 기능안전성을 위한 개발 프로세스에 적용하기에는 어려움이 있다. 특히, 해양분야 소프트웨어의 대부분은 장비와 결합한 형태의 소규모 소프트웨어이므로 NIPA에서 제시하는 공통가이드라인을 그대로 적용하는 것은 어렵다. 소프트웨어 규모보다 작성할 문서의 양이 너무 많고, 각 문서에 포함된 모든 항목을 작성하기에는 전체 개발 기간에 비해 문서화에 사용하는 부담이 실무자들에게 과도하다. 실무자가 각 산업 분야에 기능안전성 확보를 위해 가이드라인을 적용하려면 산업 분야의 특성을 반영하는 작업이 필요하다.

본 논문에서는 해양분야의 소규모 소프트웨어를 대상으로 소프트웨어 안전성 공통 개발 가이드라인을 적용하기 위한 분석을 하고, 절차와 산출물을 정의하였다. Table 3.1은 기존의 소프트웨어 개발 프로세스, 소프트웨어 안전성 공통 개발 가이드라인, 본 논문에서 정의한 프로세스를 비교·분석한 표이다.

Table 3.1 Comparative analysis table by process

구분 \ 프로세스	해양분야 소프트웨어 개발 프로세스	소프트웨어 안전성 공통 개발 가이드라인	본 논문에서 제시하는 개발 프로세스
적용대상	해양분야 소프트웨어	모든 산업분야와 다양한 규모의 소프트웨어	해양분야의 소규모 소프트웨어
기능안전성	품질의 부특성(안전성)으로만 명시	위험분석을 추가하여 품질 확보	위험분석을 간소화하여 확보
적용 절차	모든 소프트웨어 개발에 적용 가능한 절차	모든 산업분야 및 규모에 적용 가능한 절차	해양분야의 소규모 소프트웨어만을 대상으로 하는 절차
산출물	기능안전성을 포함한 많은 품질속성을 확보해야하기 때문에 다양한 산출물 목록과 템플릿 정의	산업분야와 규모에 맞게 필요한 산출물 목록이 너무 다양하고 전체 프로젝트에 적용하기 위한 내역 명시	소규모 소프트웨어를 대상으로 하는 명확한 산출물 목록과 템플릿 정의
대표 결과물	IMO MSC.1/ Circ.1512	SW 안전성 공통 개발 가이드라인	

3.1 위험분석

본 절에서는 해양분야의 소규모 소프트웨어를 대상으로 위험분석의 절차를 제시하고 필요한 고려사항들을 예시와 함께 설명한다.

3.1.1 위험분석의 고려사항

소프트웨어는 자체만으로는 안전성 문제를 발생시키지 않고 하드웨어와 결합한 시스템에서부터 발생한다. 소프트웨어의 위험분석은 전체 시스템의 설계맥락에서 수행되어야 하며 소프트웨어의 안전성은 하드웨어, 주변 환경, 사람 등을 고려해야 한다. 소프트웨어의 위험을 분석하기 위해서는 시스템 안전 기능의 수행과 시스템 제어 및 감시 기능의 수행에 있어서 소프트웨어의 역할을 이해해야 하고 해당 시스템 안전성 달성에 있어서 소프트웨어가 시스템에 미치는 영향을 잘 파악하여야 한다.

위험분석(RA)단계에서는 시스템 설계에서 어떠한 사항이 변경되어야 할 것인지를 추가로 제시할 수 있다. 위험도의 기준을 선정하고 각 위험요소를 위험도에 따라 분류한다. 위험도 분류 기준은 HAZOP 프로세스를 통해 도출되어야 하며 프로젝트의 크기에 따라 분류 기준은 변경될 수 있다. 위험분석 절차는 Fig. 3.1과 같다.

소프트웨어 요구사항 위험분석에서는 소프트웨어 요구사항 명세서가 시스템 위험요소에 미칠 영향을 조사한다. ‘소프트웨어 안전성 공통 개발 가이드’에서는 요구사항 위험분석에서 고려되어야 할 품질 속성들로 정확도 (Accuracy), 용량 (Capacity), 기능성 (Functionality), 신뢰도 (Reliability), 강인도 (Robustness), 안전성 (Safety), 보안 (Security)등을 제시하였으며 각 품질 속성들을 적용하기 위해서는 분야의 시스템에 맞게 수정하여 사용하는 것을 권고하고 있다.

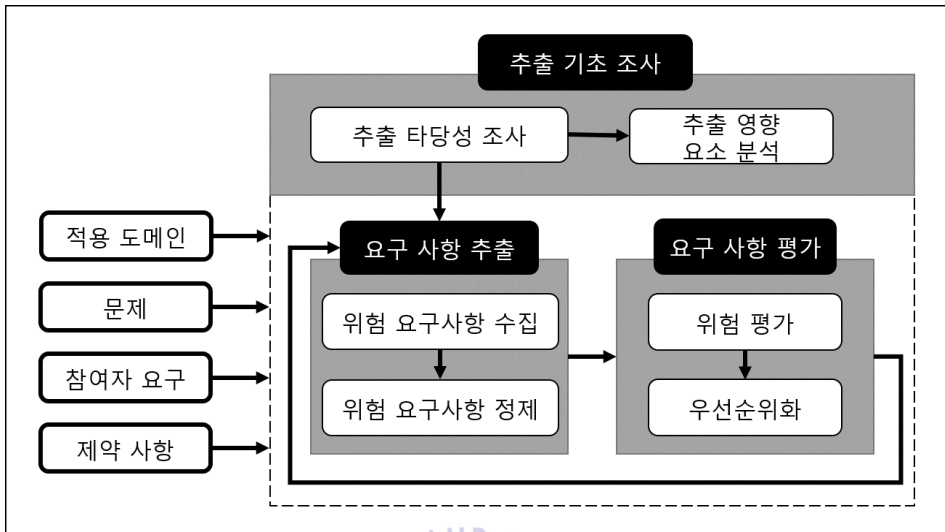


Fig. 3.1 Diagram of requirement extraction activity

3.1.2 추출 기초조사

(1) 추출 타당성 조사

추출 타당성 조사 단계에서는 인터뷰, 브레인스토밍, 프로토타이핑 등을 통해 요구사항을 추출한다. 추출된 요구사항들에 대한 제약 사항을 발견, 검토, 명확화하여 확정한다. 요구사항을 정의하기 위해서는 용어의 정의가 필수적이며 이를 안내단어 (Guideword)로 정의한다. 안내단어는 공정의 비정상적인 상태를 나타낸다. 실제 안내단어는 많은 양이 존재하며 해당 시스템에 적합한 안내단어를 HAZOP 회의 전에 여러 참가자와 함께 선택해야 한다.

(2) 추출 영향요소 분석

추출 영향요소 분석 단계에서는 확정된 요구사항에 영향을 끼칠 수 있는 요소들을 분석한다. 시나리오 분석, 작업 분석, 스토리보드, 벤치마킹, 구조적 분석, 유스케이스 기반 분석 등 다양한 분석 작업이 있다. 분석된 요구사항들은 구조화하고 이에 대한 대안을 결정하는 과정까지 포함하여야 한다.

요구사항들의 요소분석 완료 후, Table 3.1과 같이 각 안내단어의 원인을 명시하여 안내단어 리스트를 도출한다. 안내단어의 정의는 위험을 분석하기 위한 과정의 필수적인 요소이기 때문에, 산출물로 만들어 보관해야 한다. Table 3.2는 선박의 유류계를 대상으로 도출된 안내단어이다.

Table 3.2 Example template of Guideword

번호	안내단어	발생원인
1	높은 흐름	지나친 압력
2	높은 레벨	레벨조절밸브 고장
3	고압	압력조절밸브 고장
4	고온	온도조절밸브 고장
5	불순물	기체, 액체, 고체
⋮	⋮	⋮

3.1.3 요구사항 추출

(1) 위험요구사항 수집

위험요구사항 수집 단계에서는 요구사항들에 대한 정의와 요소 분석 완료 후, 요구사항에 의해 나타날 수 있는 위험요구사항을 수집한다. 인적자원, 기계적 결함 등 시나리오가 진행될 때, 나타날 수 있는 위험들을 명세화한다. 위험 요소들은 비중이 큰 위험뿐만 아니라 사소한 위험들도 막대한 피해를 줄 수 있으므로, 모든 위험요소를 포함하는 것이 중요하다. Table 3.3은 안내단어의 원인과 결과에 대한 예시이다.

Table 3.3 Example template of HAZOP worksheet

안내단어	원인	결과	안전장치
고압	압력조절밸브 고장	파이프 또는 장치 고장, 가스밸브실 및 엔진실 가스 누출, 화재 또는 폭발로 인한 치명상	수동밸브, 이중벽 파이프
저압	압력조절밸브 고장	가스공급 실패, DF engine 부하 감소	D.O supply system, 압력 송신기
불순물	파이프 내 불순 물	압력조절밸브 고장, 압력조절 실패,	
⋮	⋮	⋮	⋮

(2) 위험요구사항 분류

위험요구사항 분류 단계에서는 위험요구사항들을 위험기준에 따라 분류한다. 위험기준은 1~5 또는 상~하 등급으로 정의하며 프로젝트의 HAZOP 프로세스에서 결정된다. 각 안내단어를 대상으로 위험의 발생빈도 및 영향을 근거로 하여 위험기준에 따라 위험도를 결정한다.

해양수산부에서 발간한 ‘해양 선박사고 위기관리 표준매뉴얼’에서는 위기경보 수준을 Table 3.4와 같이 총 4가지로 분류하였다. 관심은 피해 발생 가능성이 있는 수준, 주의는 위험으로 인한 피해가 발생한 수준, 경계는 위험으로 인한 피해가 급격하게 증가하여 대규모 피해 가능성이 있는 수준, 심각은 지역적·부분적 피해 발생 및 국가적 차원에서 대처가 필요한 수준이다. 이 분류사항들은 선박의 충돌, 접촉, 침몰, 화재 및 폭발 등으로 대규모의 사상자가 발생하거나 발생이 예상되는 시스템에 적용하는 사항들이다.

Table 3.4 Degree of Hazard alert (Marine vessel accident)

분류	구분	내용
1	관심(Blue)	징후가 있으나 그 활동수준이 낮으며 가까운 기간 내에 국가위기로 발전할 가능성도 비교적 낮은 상태
2	주의(Yellow)	징후 활동이 비교적 활발하고 국가위기로 발전할 수 있는 일정 수준의 경향성이 나타나는 상태
3	경계(Orange)	징후 활동이 매우 활발하고 전개속도, 경향성 등이 현저한 수준으로서 국가위기로의 발전 가능성이 농후한 상태
4	심각(Red)	징후 활동이 매우 활발하고 전개속도, 경향성 등이 심각한 수준으로서 위기발생이 확실시 되는 상태

3.1.4 요구사항 평가

(1) 위험평가

위험평가 단계에서는 프로젝트의 위험도에 따라 분류된 위험요구사항이 적절하게 분류되었는지 확인한다. 위험평가의 주목적은 시스템, 네트워크 조직 등에서 발생할 수 있는 손실에 대비한 보안 대책에 드는 비용 효과 분석을 통해 적은 비용으로 가장 효과적인 위험관리를 수행하는 것이다. 위험이 발생했을 때, 그에 대응하는 안전조치도 명시하는 것이 효율적이다. 위험평가의 결과는 위험분석 결과서로 산출된다. Table 3.5는 유류계를 대상으로 하는 위험분석 결과서의 템플릿을 나타낸다.

(2) 우선순위화

우선순위화 단계에서는 분류된 위험요구사항을 프로젝트의 특성에 맞게 우선순위를 선정한다. 우선순위는 위험도 뿐만 아니라 프로젝트의 실행 순서, 장비와의 연결도 등을 종합적으로 고려하여 결정한다.

위험분석 프로세스의 궁극적인 목표는 부적합 사항을 찾아서 시정하고 필요한 안전 조치를 제공하는 데 있다. 소프트웨어 위험요소 분석과정에 적절한 조치가 취해지지 않는다면 분석의 의미가 없어지므로 최소한 다음 유형의 조치가 필요하다.

- 시스템 설계는 소프트웨어에 의해 영향을 받거나 소프트웨어로 적절하게 처리되지 않는 확인된 위험요소들을 제거하려는 목적으로 그 위험요소를 허용 가능한 수준까지 줄이거나 확인된 위험요소들이 심층 방어 설계로써 제거될 수 있도록 시스템 구조를 변경할 수 있다.
- 소프트웨어 설계는 확인된 위험요소들을 없애거나 그것들을 허용 가능한 수준까지 줄이려는 목적으로 변경될 수 있다.
- 소프트웨어 품질은 허용 가능한 수준까지 특정 위험요소의 발생 가능성을 줄여서 충분한 정도까지 나아질 수 있다.
- 응용 시스템은 만약 그 시스템이 너무 위험하다면 폐기될 수도 있다.

Table 3.5 Example template of Hazard analysis result

안내단어	원인	결과	안전장치	위험도	추가안전조치	비고
고압	압력조절밸브 고장	파이프 또는 장치 고장 가스밸브실, 엔진실 가스 누출 화재나 폭발로 인한 치명상	수동 밸브 이중벽 파이프	심각	가스검출기, 정지 시스템	SIL 할당
저압	압력조절밸브 고장	가스공급 실패 DF engine 부하 바로 감소	D.O supply system 압력송신기	주의		
	가스환기밸브가 잘못 열리거나 썸	가스공급실패 DF engine 부하감소 가스밸브실, 엔진실 외부에 화재 또는 폭발	D.O supply system 압력송신기	경계	가스 환기 검출기	설계정보 부족
흐름 없음	압력조절밸브 단힘 위치에 있음	가스공급 실패 MDO mode로 변화	D.O supply system 압력송신기			
불순물	파이프에 불순물	압력조절밸브 고장 압력조절 실패 단힘 위치에서 누출 DF engine 가스 투입 밸브 손상 Sol. v/v 손상 또는 폭연		경계	필터	
화재 또는 폭발	가스밸브실의 파이프에 가스 누출	화재 또는 폭발	공간 격리, GVR fan 이중벽 파이프	심각	가스검출기, 정지 시스템	SIL 할당
	엔진실의 파이프에 가스 누출	화재 또는 폭발	공간 격리, GVR fan 이중벽 파이프	심각	가스검출기, 정지 시스템	SIL 할당

3.3 소프트웨어 요구분석

본 절에서는 소규모 소프트웨어를 대상으로 요구사항 분석 단계에서의 고려사항과 절차를 제시한다. 소프트웨어 요구분석 단계는 해양 SQA/HCD 실무적용 지침의 내용을 따른다(선박해양플랜트연구소, 2015). 소프트웨어 요구분석 (SR) 단계에서는 요구사항마다 이전단계에서 선정된 위험도를 부여한다. 소프트웨어 요구분석 절차는 3.3.2에서 다룬다.

3.2.1 소프트웨어 요구분석의 고려사항

요구사항 분석에서는 이해관계자들이 모여 개발될 소프트웨어에 대한 요구사항을 논의한다. 요구사항 분석은 크게 이해관계자 인식, 요구사항 도출, 요구사항 검토 및 평가 등의 활동으로 압축할 수 있다.

3.2.2 이해관계자 인식

이해관계자 인식 단계에서는 프로젝트의 생명주기에 따라 시스템에 속한 개인 또는 합법적인 이해관계를 가진 이해관계자 그룹을 인식하여야 한다. 이해관계의 정도에 따라 다음으로 분류될 수 있다.

- (1) 개발자(프로젝트 실무자): 개발 혹은 프로젝트의 실무를 담당하는 집단
- (2) 고객: 개발된 제품을 구매하는 집단
- (3) 사용자: 개발된 제품을 사용하는 집단
- (4) 후원자: 프로젝트의 성공을 위해 지원하는 집단
- (5) 기타 관련자: 프로젝트에 직접 참가하지는 않지만 영향을 주는 사내·사외의 집단

3.2.3 요구사항 도출

요구사항 도출 단계에서는 소프트웨어의 요구사항을 식별하고 도출하는 단계이다. 소프트웨어의 요구사항은 분야에 따라 구분될 수 있다. Table 3.6은 해양 SQA/HCD 실무적용지침에서의 소프트웨어 관련 요구사항을 나타낸다.

Table 3.6 List of Requirements related software

요구사항	포함되는 내용
기능 요구사항	<ul style="list-style-type: none"> 소프트웨어가 반드시 수행해야 하는 기능들에 대한 요구사항 주 업무별 기능 요구사항을 도출하고 세부 기능에 대한 요구사항을 작성
성능 요구사항	<ul style="list-style-type: none"> 처리속도 및 시간, 자원 효율성 등에 대한 요구사항
인터페이스 요구사항	<ul style="list-style-type: none"> 소프트웨어의 인터페이스에 관련된 요구사항 화면의 크기, 배치, 구성 등을 작성
데이터 요구사항	<ul style="list-style-type: none"> 초기자료 구축, 데이터 변환, 보안이 필요한 데이터에 관한 요구사항
보안 요구사항	<ul style="list-style-type: none"> 정보 자산의 기밀성, 무결성을 확보하기 위한 요구사항 보안 메커니즘 알고리즘 등에 대한 요구사항을 작성
품질 요구사항	<ul style="list-style-type: none"> 품질에 관한 요구사항 ISO 9126/25000 시리즈의 품질 특성들을 이용하여 각 특성에 대한 요구사항을 작성

3.2.4 요구사항 상세내역 작성

요구사항을 효과적이고 지속적으로 관리하기 위해 일정한 서식에 맞추어 요구사항 상세명세를 작성하고 기록하는 작업이 필요하다. 산출물은 이후에 진행될 단계의 입력으로 작용하기 때문에 이를 취합하여 기록하는 것은 매우 중요하다. Table 3.7은 e-Navigation SQA/HCD 가이드라인에서 제시하는 요구사항 액티비티의 산출물 중 하나인 요구사항 명세서에서 각 요구사항의 위험도를 함께 명시해줄 수 있는 항목을 나타낸다. HAZOP 프로세스의 결과를 이용해 각 요구사항에 대한 위험도를 도출해낼 수 있다.

Table 3.7 Item list of Requirement specification

항목 이름		항목 내용
① 요구사항 고유번호		요구사항 추적관리를 위해 독립적인 고유번호부여
② 요구사항 명칭		요구사항 명칭을 작성
③ 요구사항 분류		요구사항 분류기준에 따른 분류 작성
요구사항 상세설명	④ 정의	요구사항 정의
	⑤ 세부내용	요구사항의 구체적인 세부내용 작성
⑥ 산출정보		해당 기능을 통해 산출되는 결과물 혹은 정보 작성
⑦ 관련 요구사항		정의된 요구사항과 관련된 요구사항에 대하여 작성
⑧ 요구사항 출처		기능 도출 내용에 대한 출처 표기
안전관련	⑨ 위험도	해당 요구사항이 갖는 위험도
	⑩ 안전책임자	요구사항의 안전에 책임을 질 수 있는 이해관계자 명시
	⑪ 위험발생원인 및 안전조치	요구사항에서 위험이 도출될 수 있는 원인을 제시하고, 관련 안전 조치를 명시

Table 3.8은 요구사항 명세서의 템플릿을 나타낸다. 요구사항 명세서는 소프트웨어의 요구사항마다 작성되어야 한다. Table 3.7의 각 항목의 번호에 맞게 다수의 요구사항 명세서가 산출물로 도출된다. 예를 들어, ① 번 항목인 요구사항 고유번호를 Table 3.8의 ① 번 자리에 기재한다. 이하 다른 항목도 같은 방법으로 작성한다. 요구사항명세서의 작성 목적과 작성 방법은 다음과 같다.

[작성 목적]

요구사항 명세서에 기록된 내용을 검토하여 해당 요구사항이 설계에 반영될 수 있는지 판단하고 반영이 어려우면 적절한 사유를 작성한다. 위험도는 위험도 매트릭스의 수준을 참고하여 결정하고 안전 책임자를 선정하여 안전성을 보장할 수 있도록 한다.

[작성 방법]

모든 요구사항에 대하여 이해관계자들이 모여 검토하며, 위험도는 HAZOP 프로세스에서 나타난 결과를 이용하여 명시한다. 요구사항이 수용되지 못한 경우에는 수용 불가사유를 적는다.

Table 3.8 Template of Requirement specification

REQ-##	요구사항 명세서				
시스템명		서브시스템명			
①	②				③
위험도	⑨	안전책임자	⑩	관련 요구사항	⑦
정의	④				
세부내용	⑤				
위험발생원인 및 안전조치	⑪				
산출 정보	⑥				
요구사항 출처	⑧				

제 4 장 사례 연구

본 장에서는 3장의 신뢰·안전성 소프트웨어 개발주기를 선박시스템에 적용하여 도출되는 산출물을 제시한다. 적용 대상은 선박에 필수적으로 탑재되어야 하는 선박 시스템인 음향 측심기로 선정하였다.

4.1 선박 시스템 음향 측심기(Echo Sounder)

Fig. 4.1은 음향 측심기의 사진이다.



Fig. 4.1 Photograph of Echo sounder

음향 측심기는 메아리처럼 선저에서 소리를 내어 그것이 해저에 부딪혀서 되돌아오는 시간으로 바다의 깊이를 측정하는 기계이다. 20세기 초에 미국에서 처음으로 고안하였으며 1922년경에 프랑스의 랑지뱅(Langevin, Pierre)이 수정의 압전효과를 이용한 초음파식 음향 측심기를 개발하였다. 1935년에 영국의 헨리 휴즈 회사(현재의 켈핀 휴즈)는 자의식 송수파기와 기록지를 이용하는 음향 측심기를 개발하였다. 음향 측심기에서 송출하는 음파의 전달 속도는 바다의 온도, 염분, 수압 등에 따라 달라지므로 측정값을 주변 환경에 맞게 수정해야 한다. 음향 측심기의 출력방식은 기록지에 바늘로 수심을 표시하는 아날로그 방식과 LCD 화면에서 바로 수심이 표시되는 디지털 방식이 있다. 디지털 방식

음향 측심기는 연결된 프린터로 기록을 남길 수 있다.

Table 4.1은 ISO 9875(Ships and marine technology)를 기준으로 음향 측심기에 대한 소프트웨어 중심의 기능안전성 척도를 위해 관련된 소프트웨어 항목을 구분한 것이다. 위험분석을 위해 각 항목을 분야에 따라 소프트웨어, 하드웨어 및 운용으로 구분하였다. 세 가지로 분류된 각 항목 중 소프트웨어로 분류된 항목을 대상으로 위험분석을 한다.

Table 4.1 Performance standard of Echo sounder

분류 항목		설명	구분
①일반 사항		음향 측심기는 다음 성능 요건을 만족하여야 하며 적용 가능한 경우 IEC 60945(Maritime navigation and radiocommunication equipment and systems)의 일반 요건에도 적합하여야 한다.	운용
②기능	성능 범위	이 장치는 일상 전파와 해저 반사 성능 조건에도 송수파기로 2m ~ 200m의 변환기에서 모든 간격을 측정할 수 있어야 한다.	소프트웨어 하드웨어
	범위 눈금	이 장치는 최소 2개 눈금 범위를 갖는다. 눈금 범위는 얇은 수심 범위인 20m범위와 깊은 수심 범위인 200m범위로 분류된다. 자동 범위 기능은 분류된 범위를 자동적으로 선택한다. 위성 범위가 0부터 초기화되지 않고 사용될 경우에는 그 범위가 사용되고 있음을 보여주어야 한다.	소프트웨어 하드웨어
	주 디스플레이	주 디스플레이는 직접적인 깊이와 가시적인 측심 기록을 제공하는 적합한 그래픽을 출력한다. 주 디스플레이 기록은 깊은 범위 눈금에서 최소 15분간 측심을 나타내어야 하며 다양한 색상의 디스플레이를 사용할 수 있어야 한다.	소프트웨어

	기타 디스플레이	기타 디스플레이는 다른 모양의 디스플레이를 추가할 수 있다. 그러나 주 디스플레이의 일상적인 동작에 영향을 미치지 않아야 한다.	하드웨어 운용
	펄스 반복률	펄스 반복률은 깊은 수심 범위에서 매 분당 12펄스, 얇은 수심 범위에서 매 분당 36펄스보다 느려서는 안 된다.	소프트웨어
	횡경사 및 종경사	선박의 횡경사 ± 10 도 또는 종경사 ± 5 도일 때, 장치 성능은 이 표준의 요건에 적합하여야 한다.	소프트웨어
③	다중 설치	변환기 및 관련된 전달 매체 수신기는 1개 이상 설치할 수 있다. 1개 이상을 사용하게 된다면 서로 다른 변환기로부터 그 깊이를 각각 분리하여 출력할 수 있는 수단이 있어야 하고 사용 중인 변환기를 명확히 표시하여야 한다.	하드웨어 운용
④	자료 보관	서류 기록이나 다른 수단으로 정보(깊이 및 12시간 동안 관련된 시간)를 기록할 수 있어야 한다.	소프트웨어 운용
⑤	정확성	1500m/s의 수중 음속을 근거로 할 때, 표시된 깊이의 허용 공차는 다음 중 큰 값으로 한다. <ul style="list-style-type: none"> - 얇은 수심 범위 눈금에서는 ± 0.5m, 깊은 수심 범위 눈금에서는 ± 5m - 표시된 깊이의 $\pm 2.5\%$ 디스플레이 눈금은 얇은 수심 눈금에서 1m당 5.0mm보다 작지 않아야 하고 깊은 수심 눈금에서는 1m당 0.5mm보다 작지 않아야 한다.	소프트웨어
⑥	오동작, 경보 및 표시	경보 신호는 가시광선 또는 묵음 기능으로 수심이 초기 설정 값보다 낮을 때 제공되어야 한다. 초기 설정 경보 깊이를 변환기 위치에서 참조하지 않을 경우에는 기준 위치를 표시하여야 한다. 전원의 공급이 실패하거나 감소되는 상태를 표시하는 설비는 배전반	소프트웨어 하드웨어

	이나 그 외 장소 쪽으로 통합할 수 있으며 반드시 이 장치와 일체로 할 필요는 없다.	
⑦ 인간공학적 기준	범위 눈금 선택 기능은 사용자가 직접 접근할 수 있어야 한다. 다른 기능도 직접 접근할 수 있어야 하고, 관련 메뉴에 있는 정해진 제어나 주접근으로 즉시 실행되어야 한다. 수심 정보의 깊이는 사용 범위/눈금의 1/10 미만 간격으로, 시간은 5분을 초과하지 않는 간격으로 표시하여야 한다.	소프트웨어
⑧ 설계 및 설치	이 장치는 IMO Resolution A. 694(17)에 적합하여야 한다.	운용
⑨ 인터페이스	출력으로부터 구한 깊이 정보를 원격 디지털 디스플레이, 항해 자료 기록기 및 항적 제어 시스템과 같은 다른 장치에 제공할 수 있어야 한다. 이 출력에서는 선박용 골 하부 깊이, 현재 출력되는 깊이 눈금, 병렬(다중, 복수) 설치로 사용하는 송수파기, 적용 가능한 기타 상태 정보를 포함하여야 한다. 이 출력은 디지털 방식, 연속 전달 방식이어야 하며, 관련된 국제 기준(IEC 61162)에 적합한 설비이어야 한다.	소프트웨어
⑩ 안전 예방 조치	고전압 전기 감광지의 기록 매체를 사용하거나 유동성 기록 기구를 사용하는 경우와 음향 측심기가 동작하는 동안 그 기록으로 접근이 가능한 경우의 차이는 운전자를 위해 안전하게 공급되어야 한다.	소프트웨어
⑪ 표시	이 장치는 제조자명, 형식 및 일련번호를 표시하며 공급되어야 한다.	운용
⑫ 정보	선박과 관련된 선원이 그 장치를 효과적으로 작동하고 관리할 수 있도록 정보를 제공하여야 한다.	운용

4.2 음향 측심기의 위험분석 절차 적용

4.1절에서 작성된 음향 측심기의 성능 규격 분석 결과 중 소프트웨어로 분류된 항목을 대상으로 위험분석을 한다. Table 4.2는 음향 측심기에서 소프트웨어 항목을 정리한 것이다.

Table 4.2 Software category of Echo sounder function

분류 항목		구분
기능	성능 범위	소프트웨어, 하드웨어
	범위 눈금	소프트웨어, 하드웨어
	주 디스플레이	소프트웨어
	펄스 반복률	소프트웨어
	횡경사 및 종경사	소프트웨어
자료 보관		소프트웨어, 운용
정확성		소프트웨어
오동작, 경보 및 표시		소프트웨어, 하드웨어
인간공학적 기준		소프트웨어
인터페이스		소프트웨어
안전 예방 조치		소프트웨어

4.2.1 추출 기초조사

(1) 추출 타당성 조사

음향 측심기의 성능규격에서 소프트웨어의 위험이 발생할 수 있는 총 12개의 항목을 대상으로 요구사항을 추출한다. 모든 위험요소를 식별하기 위해서는 유지보수, 퍼지(Purge), 정비, 시동 등 모든 운용 모드에 대해 HAZOP 프로세스를 수행하여야 한다. 안내단어는 음향 측심기가 작동하지만 기능을 제대로 수행하지 못해서 생기는 위험요소(측정 오류, 범위 오류, 눈금 간격 오류 등)와 음향 측심기가 제대로 작동하지 않아 발생하는 위험요소(기록 오류, 오동작, 접근 불

가 등)가 나타난다. 본 논문에서 제시하는 안내 단어는 한국해양대학교의 한나라 호에 설치되어있는 음향 측심기의 제조회사인 (주)삼영이엔씨의 매뉴얼과 실무에 종사하고 있는 항해사들의 설문을 통해 도출되어 Table 4.3으로 나타난다.

Table 4.3 Guideword list of Echo sounder

번호	안내단어
1	측정 오류
2	범위 오류
3	눈금간격 오류
4	기록 오류
5	다중 정보
6	자료 손실
7	정확성 오류
8	오동작
9	경보 신호
10	접근 불가

(2) 추출 영향요소 분석

확정된 요구사항에 영향을 끼치는 요소를 분석한다. 분석이 완료된 후에는 Table 4.4와 같이 안내단어의 발생 원인을 명시한다. 안내단어의 발생 원인은 성능규격 및 실무에서 사용하고 있는 음향 측심기의 매뉴얼을 대상으로 조사한 결과이다.

Table 4.4 Guideword of Echo sounder

번호	안내단어	발생원인
1	측정 오류	일상 전파와 해저 반사 실패
2	범위 오류	수심 범위가 너무 낮거나 높음
3	눈금간격 오류	범위 눈금 미사용
4	기록 오류	디스플레이 출력 실패
5	다중 정보	너무 많은 정보 수집 및 변환실패
6	자료 손실	서류 및 정보 손실
7	정확성 오류	깊이의 허용 공차의 초과
8	오동작	기기의 동작 오류
9	경보 신호	수심이 초기 설정보다 낮음
10	접근 불가	권한 손실 및 인터페이스 오류

4.2.2 요구사항 추출

(1) 위험요구사항 수집

위험요구사항 수집단계에서는 요구사항에 의해 나타날 수 있는 위험요구사항을 수집하는 단계이다. Table 4.5는 Table 4.4에서 제시한 각 안내단어로 인해 나타날 수 있는 결과와 위험을 해결하기 위한 안전장치를 나타낸 표이며 결과를 토대로 위험요구사항을 수집한다.

(2) 위험요구사항 분류

3.2절에서 제시한 위기경보 단계는 선박이 충돌, 접촉, 좌초, 침몰, 화재 및 폭발 등으로 대규모의 사상자가 발생하거나 발생이 예상되는 범위에서의 위기경보 수준을 나타낸다. 이는 ‘재난 및 안전관리 기본법 제34조 5항 위기관리 매뉴얼’에서 정하는 바에 따라 4개의 단계를 말한다. 해양분야의 다양한 관점에서 안전사항은 대부분 이 4단계의 수준을 가지며 수준에 대한 내용을 규모에 맞게 수정하고 있다. 본 논문에서는 선박에 탑재된 소규모 장비들을 대상으로 위험분석을 시행하기 때문에 Table 4.6과 같이 위험평가 기준을 정의하였다.

Table 4.5 HAZOP worksheet of Echo sounder

안내단어	원인	결과	안전장치
측정 오류	일상 전파와 해저 반사 실패	수심 측정 불가	송수파기 제어
범위 오류	수심 범위가 너무 낮거나 높음	수심 결과 오류 신뢰성 없는 결과도출	수동범위 제어 시스템
기록 오류	디스플레이 출력 실패	기기 사용 불가	
정확성 오류	깊이의 허용 공차의 초과	신뢰성 없는 결과도출	
다중 정보	너무 많은 정보 수집 및 변환실패	신뢰성 없는 결과도출	변환기
자료 손실	서류 및 정보 손실	항해정보 누락	백업 DB
오동작	기기의 동작 오류	기기 사용 불가	
접근 불가	권한 손실 및 인터페이스 오류	기기 사용 불가	
경보	수심이 초기 설정보다 낮음	좌초 또는 바텀터치	경보 신호

Table 4.6 Degree of Hazard alert (Maritime equipment software)

분류	구분	내용
1	관심 (Blue)	징후가 있으나 그 활동수준이 낮으며 가까운 기간 내에 소프트웨어 위험으로 발전할 가능성도 비교적 낮은 상태
2	주의 (Yellow)	징후 활동이 비교적 활발하고 소프트웨어 위험으로 발전할 수 있는 일정 수준의 경향성이 나타나는 상태
3	경계 (Orange)	징후 활동이 매우 활발하고 전개속도, 경향성 등이 현저한 수준으로서 소프트웨어 위험으로의 발전 가능성이 농후한 상태
4	심각 (Red)	징후 활동이 매우 활발하고 전개속도, 경향성 등이 심각한 수준으로서 소프트웨어 위험으로 발생이 확실시 되는 상태

4.2.3 요구사항 평가

(1) 위험평가

Table 4.7은 음향 측심기에 대한 HAZOP 프로세스의 산출물이다. 음향 측심기의 안내단어 중 9개의 위험요소를 식별하여 나열한 것이다. 4개는 관심, 3개는 주의, 2개는 경계 수준의 위험도를 가지고 있다. 다음은 위험요소들에 대한 안전조치들을 나타낸다.

① 트랜듀서 확인: 음향 측심기는 선박의 하단부에 부착된 트랜듀서에서 송출하는 음파를 통해 수심을 측정한다. 트랜듀서의 고장은 수심측정의 가장 큰 원인이 될 수 있다.

② 수동 설정: 음향 측심기는 2m~200m의 변환기에서 모든 간격을 측정할 수 있어야 한다. 이 범위에서 벗어날 경우에 수동 설정을 통해 오류를 해결할 수 있어야 한다.

③ 케이블 및 설정 확인: 소프트웨어를 제공하는 하드웨어의 전원공급을 담

당하는 케이블, 또는 데이터를 제공하는 커넥터의 연결 상태를 확인한다.

④ 변환기 설정 변경: 변환기 및 이와 관련된 전달 매체 수신기가 다수일 경우, 각 변환기의 상태 또는 설정을 관리할 수 있어야 한다.

⑤ 데이터 복구: 데이터가 잘못되었을 경우, 자료를 백업해놓은 시점으로 돌아간다. 데이터의 백업 주기는 실무자들의 회의를 통해 결정한다.

⑥ 권한 획득 및 리셋: 권한이 손실되었을 경우, 제조사와 설계팀에 의뢰하여 권한을 획득하거나 시스템을 리셋 한다.

⑦ 항해 경로 변경: 초기설정이란, 선박이 좌초될 위험이 있는 수심으로 설정한다. 만약, 수심이 이 설정 값보다 낮으면 항해 경로를 변경한다.

(2) 우선순위화

분류된 위험요구사항의 위험도를 고려하여 우선순위를 부여한다. 각 요구사항에 부여된 우선순위를 소프트웨어 개발에 적용한다.



Table 4.7 Safety-conscious HAZOP Worksheet of Echo sounder

안내어	원인	결과	안전장치	위험도	추가안전조치	비고
측정 오류	일상 전파와 해저 반사 실패	수심 측정 불가	송수파기 제어	관심	트랜듀서 확인	
범위 오류	수심 범위가 너무 낮거나 높음	수심 결과 오류 신뢰성 없는 결과도출	수동범위 제어 시스템	관심	수동 설정	
기록 오류	디스플레이 출력 실패	기기 사용 불가	보조 기기 사용	주의	케이블 및 설정 확인	
정확성 오류	깊이의 허용 공차 초과	신뢰성 없는 결과도출	수동 제어 시스템	관심	트랜듀서 확인	
다중 정보	너무 많은 정보의 수집 및 변환 실패	신뢰성 없는 결과도출	변환기	관심	변환기 설정 변경	
자료 손실	서류 및 정보 손실	항해정보 누락	백업 DB	주의	데이터 복구	
오작동	기기의 작동 오류	기기 사용 불가	보조 기기 사용	경계		
접근 불가	권한 손실 및 인터페이스 오류	기기 사용 불가	보조 기기 사용	경계	권한 획득 및 리셋	
경보	수심이 초기설정 보다 낮음	좌초 또는 바텀터치	경보 신호	주의	항해경로 변경	선박 좌초 경보

4.3 음향 측심기의 소프트웨어 요구분석 절차 적용

본 절에서는 음향 측심기를 대상으로 요구분석 단계의 절차를 적용한다.

4.3.1 이해관계자 인식

음향 측심기의 개발에 관련된 공통된 이해관계자는 다음과 같이 나타난다.

- ① 개발자(프로젝트 실무자): 음향측심기를 만드는 제조사
- ② 고객: 선박회사 및 해양 관련 종사자
- ③ 사용자: 선박의 항해사
- ④ 후원자 및 ⑤ 기타 관련자는 프로젝트에 따라 상이하며 본 논문에서는 작성하지 않는다.

4.3.2 요구사항 도출

음향 측심기는 본체와 트랜듀서 2가지의 서브 시스템으로 분류되며 각 서브 시스템마다 요구사항이 나타난다. 요구사항은 프로젝트에 맞게 분석되어야 하며, 요구사항 분석의 산출물은 요구사항 명세서로 나타난다.

4.3.3 요구사항 명세서 작성

요구사항 명세서는 전체 시스템의 개략적인 내용을 보여주는 것을 목적으로 하며 시스템의 전체적인 구성과 컴퓨터 시스템과 외부 장치 간의 인터페이스를 중점으로 기술하여야 한다. 본 논문에서는 모든 요구사항이 아닌 위험도와 관련이 있는 요구사항을 대상으로 명세서를 작성하였다. Table 4.8은 본체를 대상으로 한 기능 요구사항의 명세서이다.

Table 4.8 Template of Requirement specification of Echo sounder
(Main body)

REQ-DP	요구사항 명세서				
시스템명	음향 측심기	서브시스템명	본체		
01	설치 및 동작			분류	기능 요구사항
위험도	경계	안전책임자	설계 팀	관련 요구사항	REQ-XX-YY REQ-XX-YY
정의	본체에 전원을 공급하며 초기설정을 위한 설치를 실시한다.				
세부내용	<p>전원이 취약한 선박에서 사용하기 위해서 음향 측심기의 입력전원은 낮은 전압으로도 동작할 수 있어야 한다.</p> <p>긴급한 상황에서도 전원을 공급받을 수 있는 보조배터리를 사용할 수 있어야 하며 어느 곳이든 운용이 가능할 수 있어야 한다.</p> <p>기기에 문제가 발생하였을 경우, 문제를 해결할 수 있는 엔지니어와 소통이 원활하게 이루어져야 한다.</p>				
위험발생원인 및 안전조치	<p>① 기기에 전원을 공급하고도 오작동을 할 경우에는 설계팀에 문의하여 문제를 해결하거나 새로운 장비로 대체하여야 한다.</p> <p>② 소프트웨어의 오작동으로 인해 사용자가 음향 측심기를 사용할 수 없을 경우에도 위와 같은 해결방법을 실시한다.</p>				
산출 정보	음향 측심기 설치 매뉴얼, 설치 시나리오				
요구사항 출처	REQ-ZZ				

02	디스플레이 출력			분류	기능 요구사항
위험도	주의	안전책임자	설계 팀	관련 요구사항	REQ-XX-YY REQ-XX-YY
정의	사용자가 측정된 수심을 가시적으로 확인할 수 있도록 디스플레이에 출력한다.				
세부내용	<p>수심에 대한 데이터를 수집한 후, 화면 또는 기기 내에 설치된 기록용지에 출력한다. (용지 사이즈는 000mm의 0.0인치로 고정한다.)</p> <p>사용자가 설정하는 동안에도 디스플레이에는 수심데이터를 항상 표시하여야 한다. (수심 데이터 출력과 파라미터 정보출력을 구분하여야 한다.)</p> <p>사용자는 항상 디스플레이의 내용을 인지하여야 한다. (주변의 밝기에 상관없이 가독성이 높아야 한다.)</p>				
위험발생원인 및 안전조치	<p>① 디스플레이에 수심이 나타나지 않을 경우, 기기의 전원과 데이터 수집을 담당하는 케이블 및 설정을 확인하여야 한다.</p> <p>② 변환기 및 수신기가 다수일 경우, 정보의 수집 및 변환에 실패하여 신뢰할 수 없는 결과가 도출될 수 있기 때문에 각 변환기들을 빠르게 제어할 수 있어야 한다.</p>				
산출 정보	음향 측심기 동작 시나리오				
요구사항 출처	REQ-ZZ				

03	수심데이터 저장			분류	기능 요구사항
위험도	주의	안전책임자	설계 팀	관련 요구사항	REQ-XX-YY REQ-XX-YY
정의		측정된 수심데이터들을 데이터베이스에 저장한다.			
세부내용		선박이 항해하는 항로의 수심데이터들을 데이터베이스에 저장하여 동일한 항로를 이용하는 선박들에게 제공한다. 신뢰성이 높은 데이터를 미리 제공받음으로써 안전한 항해를 할 수 있도록 한다.			
위험발생원인 및 안전조치		① 수심데이터들을 손실하였을 경우, 기존 데이터베이스에 저장되어 있는 수심데이터들을 이용하여 복구할 수 있도록 한다.			
산출 정보		데이터베이스			
요구사항 출처		REQ-ZZ			

04	좌초 경보			분류	기능 요구사항
위험도	주의	안전책임자	설계 팀	관련 요구사항	REQ-XX-YY REQ-XX-YY
정의		선박의 좌초 또는 바텀터치를 할 수 있는 위험상황일 경우, 경보를 울려 선원들에게 알린다.			
세부내용		측정된 수심데이터를 분석하여 예상경로의 수심을 예상할 수 있어야 한다. 예상경로가 음향 측심기에 설정된 선박하단부 높이보다 낮을 경우, 선박의 좌초 또는 바텀터치를 할 수 있기 때문에 경보를 울려 선원들에게 위험상황을 인지할 수 있도록 한다.			
위험발생원인 및 안전조치		항해변경을 권고하는 경보를 울려 선박의 좌초 또는 바텀터치가 일어날 수 있는 상황을 예방할 수 있도록 한다.			
산출 정보		좌초 경보 대응 시나리오			
요구사항 출처		REQ-ZZ			

⋮

Table 4.9는 트랜듀서를 대상으로 한 기능 요구사항의 명세서이다.

Table 4.9 Template of Requirement specification of Echo sounder(Transducer)

REQ-TR		요구사항 명세서			
시스템명	음향 측심기	서브시스템 명	트랜듀서		
01	수심 측정			분류	기능 요구사항
위험도	관심	안전책임자	설계 팀	관련 요구사항	REQ-XX-YY REQ-XX-YY
정의	트랜듀서에서 방출하는 음파가 해저에서 반사되어 돌아오는 시간을 계산하여 수심을 측정한다.				
세부내용	해저면에 따라 음파의 펄스폭과 파장을 조절하여 효율적인 측정을 실시하여야 한다. 갯벌이나 진흙같이 흡수율이 좋은 해저면에서는 펄스폭을 두껍게 하는 것이 효율적이다.				
위험발생원인 및 안전조치	① 전파의 송수신 오류 또는 해저 반사를 실패하였을 경우, 트랜듀서의 송수파기 점검을 실시하여야 한다. ② 자동범위 제어 시스템의 오작동 또는 깊이의 허용 공차가 초과될 경우에는 신빙성이 없는 결과가 도출될 수 있다. 이를 예방하기 위해 수동범위 제어 시스템 및 수동 제어 시스템이 내장되어 있어야 한다.				
산출 정보	음향 측심기 동작 시나리오				
요구사항 출처	REQ-ZZ				
⋮					

제 5 장 결론 및 향후과제

기능안전성은 2000년 초반부터 항공, 의료, 철도 등 여러 분야에서 고려되고 있는 품질 속성 중 하나이다. 하지만 해양분야에서는 기능안전성을 만족하기 위한 대책이 마련되어있지 않다. 본 논문에서는 기능안전, 신뢰성, 품질 및 성능에 대한 검증을 위해 IEC 61508을 기반으로 하여 소프트웨어 안전성 공통 개발 가이드라인의 필요성을 다루었으며 기존의 IMO SQA/HCD 실무적용지침에 기능안전성 개념을 도입하여 요구사항 단계에서부터 생애주기 동안 관리할 수 있는 기반을 마련하였다.

본 논문에서는 HAZOP 프로세스를 통한 위험도 매트릭스를 작성하고 이를 산출물에 명시하여 안전성에 대한 개념과 수치를 명확히 하였다. 요구사항 명세서에서 해당 위험도를 나타내어 이후에 진행되는 설계, 구현, 테스트 및 운용 단계에서도 기능안전성을 확보하였다. 산출물의 템플릿과 작성방법, 목적을 제시하였지만 가이드라인 문서의 특성상 실무에 바로 적용하기 어려우므로 음향 측심기를 대상으로 한 사례 연구를 하여 실무자들의 이해를 돕는다.

본 논문에서는 요구사항 단계에서의 기능안전성을 만족하기 위한 방법을 제시하였다. 기능안전성은 프로젝트의 전체 생애주기에서 고려되어야 하기 때문에 향후 전체 생애주기에 기능안전성을 적용하는 방법을 마련하는 연구가 필요하다. 또한, 음향 측심기뿐만이 아닌 사고와 연관될 수 있는 추가적인 사례연구들이 진행되어야 하며 이를 참고하여 항해 장비들을 개발할 때 사고에 대한 인식을 넓히고 빠른 대처를 할 수 있어야 한다.

참고문헌

- Albert G. Reitan, 1998, Specialist's factual report of investigation - DCA97MA058.
- IEC 61508, 2010. - Functional safety of electrical/electronic/programmable electronic safety-related systems. IEC Publication.
- IMO., 2015, Guideline on software quality assurance and human-centred design for e-navigation. MSC.1/Circ.1512
- Kim, H., 2015. A Survey on the status of Marine IT Industrial environment for e-Navigation SQA - focusing on Korean domestic companies. *International Conference of Advanced Intelligent Maritime Safety and Technology*
- Lee, S., Lemon, N. & Lutzhoft, M., 2015. Harmonizing Guidance for Future Ship Navigation Systems Developing Guideline for Software Quality and Human-Centered Design. *Sea Technology*, 56(11), pp.41-44.
- NIPA(정보통신산업진흥원), 2016. *SW 안전성 공통 개발 가이드*, 충청도: 정보통신산업진흥원.
- Nancy Leveson, 1995, Medical Devices: The Therac-25.
- National Transportation Safety Board (NTSB), 2009. *eRailroad Accident Report - Collision of Two Washington Metropolitan Area Transit Authority Metorail Trains Near Fort Totten Station Washington, D.C.*, Washington, D.C.: NTSB.
- 김명희, 박만곤, 2012. 소프트웨어 안전성 평가를 위한 소프트웨어 고장 유형과 영향 분석에 관한 연구. *한국멀티미디어학회논문지*, 15(1), pp.115-130.

선박해양플랜트연구소, 2015. *시스템 품질인증체계 개발, 법률영향분석 및 국제 공동시험지원*: 선박해양플랜트연구소.

소프트웨어정책연구소, 2016. *소프트웨어 안전성 확보 체계에 관한 연구*, 경기도: 소프트웨어정책연구소.

이장수, 이동아, 2015. 소프트웨어 기반의 안전 필수 시스템을 위한 안전성 분석 기법. *정보과학학회지*, 33(7), pp.41-46.

한국방송통신전파진흥원, 2013. *IT 융합 산업의 H/W 및 S/W의 안전표준화 기술 동향*, 서울시: 한국방송통신전파진흥원.

