

Technical Paper

DOI: <https://doi.org/10.6108/KSPE.2021.25.3.127>

## 자동화 검증시스템을 이용한 발사관제시스템 알고리즘 검증시험

안재철<sup>a,b</sup> · 문경록<sup>a</sup> · 오일석<sup>b,\*</sup>

### The Verification Test of Launch Control System Algorithms Using Automated Verification System

Jae-Chel An<sup>a,b</sup> · Kyung-Rok Moon<sup>a</sup> · Il-Seok Oh<sup>b,\*</sup><sup>a</sup>Launch Complex Team, Korea Aerospace Research Institute, Korea<sup>b</sup>Division of Computer Science and Engineering, Chonbuk National University, Korea

\*Corresponding author. E-mail: isoh@jbnu.ac.kr

#### ABSTRACT

The launch complex(LC) is composed of various facilities. The launch control system that operates remotely those of LC spends much time and labor for developing and verifying its control algorithms. The verification of algorithms is performed by the software developer entering simulated state values based on the test procedure and checking the output result according to the algorithm flow. These verification processes should be performed repeatedly, thus the human errors are easily occurred. In this paper, an efficient automated verification method with a script test procedure is proposed to minimize human errors and shorten the verification duration. We also present the results of the algorithm verification tests for the cases of the compressed gases supply system and the electro pneumatic panel system of LC.

#### 초 록

발사대시스템은 다양한 설비로 구성된다. 이들 설비를 원격 운용하는 발사관제시스템은 제어 알고리즘 개발과 검증에 많은 시간과 노력이 소요된다. 알고리즘 검증은 소프트웨어 제작자가 시험절차서에 기반한 시뮬레이션 상태 값을 입력하고, 알고리즘 흐름에 따라 출력 결과를 확인하는 과정을 통해 이루어진다. 이러한 검증 과정은 반복적으로 수행되므로 작업 오류를 쉽게 유발할 수 있다. 본 논문에서는 작업 오류의 최소화와 검증 기간 단축이 가능한 스크립트 시험절차서 기반의 효율적인 자동화 검증 방법을 제안하며, 발사대의 고압가스 공급설비와 전자식 유공압패널설비 제어 알고리즘 검증시험에 적용한 결과를 제시한다.

Key Words: Control Algorithms(제어 알고리즘), Launch Complex(발사대시스템), Launch Control System(발사관제시스템), PLC(Programmable Logic Controller)

#### 1. 서 론

Received 27 January 2021 / Revised 10 June 2021 / Accepted 15 June 2021

Copyright © The Korean Society of Propulsion Engineers  
pISSN 1226-6027 / eISSN 2288-4548

발사대시스템은 다양한 설비와 이를 제어하기 위한 PLC 시스템들로 구성된다. PLC 시스템은

실제 운용 전 탑재 프로그램의 정확성을 검증하여 오작동으로 인한 피해를 최소화해야 한다. 전통적인 PLC 운용 프로그램 검증 절차와 방법은 많은 시간과 인력의 투입이 요구되며 시험 과정에서 검증 지연요인이 발생된다. 대표적인 검증 지연요인으로 설계 플로우차트 작성 오류, 프로그래머의 코딩 오류, 시험 절차서 오류, 시뮬레이션 오류와 장시간 시험과 피로 누적에 따른 작업 오류(Human Error) 등을 꼽을 수 있다[1]. 특히, 시험 절차서에 따른 시뮬레이션 입출력 검증은 장시간 작업자의 수작업에 의존하기 때문에 다양한 작업 오류를 발생시키기 쉽다.

본 논문에서는 작업 오류 최소화, 검증시간 단축, 체계적인 시험절차와 방법을 규정하기 위한 스크립트 시험절차 기반의 자동화 검증 방법을 제안한다. 이 방법을 고압가스공급설비와 전자식 유공압패널설비 제어시스템에 적용, 수작업 시험절차 대비 약 60~80% 단축된 검증시간 결과를 얻을 수 있었다.

2장에서 검증시스템 개발 배경과 연구동향, 3장에서 발사관제시스템 PLC 프로그램의 설계, 구현 및 검증 방법을 소개하고 개선 방안 제시와 구현 방안을 설명한다. 4장에서는 검증 대상에 대한 설명과 검증시스템을 활용한 검증시험 결과를 제시한다.

## 2. 연구 배경 및 동향

PLC 시스템은 산업설비인 자동차, 철도, 원자력 발전분야와 항공 우주 및 방위산업 분야에도 널리 사용되므로 PLC 제어 프로그램 오류로 인한 오작동은 인적 물적 자원에 직접적인 영향을 미친다. 따라서 엄격한 품질과 안전 기준이 적용된다. IEC61508은 모든 종류의 산업에 적용 가능한 기능 안전 표준이며 이를 모 표준으로 많은 분야에서 각기 부합되는 방향으로 소프트웨어의 안전 표준을 규정하고 있다.

C나 C++과 같은 범용 개발언어를 사용하는 임베디드 시스템 분야는 안전 표준을 검증하기 위한 다양한 정적/동적 검증 도구들도 쉽게 찾

아 볼 수 있다[2]. 반면, PLC 시스템 분야는 시스템을 제공하는 업체마다 다른 환경의 소프트웨어 개발 툴과 개발 언어의 상이한 함수 및 코딩 규칙을 제공한다. 이는 제조사간 프로그램의 호환뿐만 아니라 코드의 범용적 활용이 불가능한 문제를 내포하고 있다. 이에 PLC 프로그램 검증을 위한 범용 정적/동적 검증 툴은 존재하지 않으며, 업체 별 제공되는 개발 툴에 속한 기능에 의존적이다. 따라서 보조적인 검증 툴 또는 검증 시스템이 필요하다.

프로그램의 검증 방법은 크게 시험 시나리오(Test Case) 기반의 '시뮬레이션 검증 방법'과 모델검증(model checking) 기반의 '정형 검증 방법'으로 구분할 수 있다. 시뮬레이션 검증 방법은 얼마나 많은 시나리오를 설정하여 반복 시험을 수행 하였는가에 따라 프로그램의 안전성을 확보하는 요인으로 작용하나 시나리오 설정에 한계가 있어 프로그램의 안전성을 완벽하게 보장할 수는 없다. 이를 보완한 정형 검증 방법 연구가 학술적으로는 더 많이 진행되고 있으나, PLC 프로그램을 모델링 언어로 변환 및 명세작성의 어려움으로 모델검증 기반 정형 검증 방법은 아직까지도 산업 현장 도입에 걸림돌이 되고 있다.

시뮬레이션 검증에 사용되는 시뮬레이터 구성 방법에는 PLC 프로그램 입출력에 대한 응답 로직을 PC 기반 프로그램 시뮬레이터로 구현 활용하는 방법[3], 프로그램과 하드웨어 입출력 시뮬레이터를 구현 활용하는 방법[4-7], 자동화 생산 설비 분야에서 3D 기반 가상의 설비를 시뮬레이터로 구현하고 설비 간 동작 간섭 등을 사전 확인하는 방법[8-11] 등이 시도되고 있다.

모델검증은 프로그램을 유한상태기계(FSM, Finite State Machine)의 형태로 설계명세(Design Specification)하고 시스템 요구명세(Requirement Specification)를 LTL(Linear Temporal Logic)이나 CTL(Computational Tree Logic)로 표현하여 요구명세 조건이 설계명세(Design Specification)된 시스템에서 만족하는지 검증하는 방법으로 SMV, NuSMV, Cadence SMV, VIS, Uppaal 등과 같은 다양한 모델검사기(Model Checker)가

사용된다.

모델검증은 다음과 같은 세 가지 문제를 내포하고 있다. 첫째, PLC 프로그램은 모델검사의 입력 언어로 동일하게 표현될 수 없다. 둘째, 규모가 큰 프로그램은 상태 폭발(State Explosion Problem[23])이라는 제한적 현상이 발생한다. 셋째, 반례로 제시된 검증 결과를 PLC 프로그램 수정에 적용하기 어렵다. 이러한 이유로 모델검사를 활용한 정형검증은 사용에 한계를 갖는다. 상기 문제를 해결하기 위해 PLC 프로그램을 정형 모델로 동일하게 표현하기 위한 변환 연구[12-19]나 상태 공간의 축소를 위한 연구[16,20,21] 또는 반례로 제시된 결과를 직관적이고 효과적으로 분석하기 위한 연구가 시도되고 있다[22].

본 논문에서는 시뮬레이터 없이 사용 가능하며, 수작업에 의존해온 부분을 자동화하고, 모델링을 통한 정적 검증방법 대신 제어기 탑재코드를 동적 검증방법을 사용하여 검증하는 방법을 제시한다. 이는 산업용 Modbus TCP-IP 통신의 사용이 가능한 제어기에 쉽게 적용 및 활용이 가능하여 다양한 PLC 시스템 검증에 바로 사용할 수 있다.

### 3. 프로그램 검증과 제안 방향

본 절에서는 발사관제시스템용 PLC 프로그램 검증을 위한 시험절차와 방법을 가스저장 용기를 운반하는 탱크로리로부터 헬륨가스 보관용 탱크에 저장하는 가상 상황을 예로 설명하고 이 과정에서 도출된 문제점과 개선 방안을 제시한다.

#### 3.1 검증 절차와 방법

헬륨을 저장탱크로 공급하기 위한 단계를 다음과 같이 구분하였다(Figs. 1-3).

- 충전 초기단계: 탱크로리로부터 헬륨이 배관에 공급되어 있지 않아 밸브 전단에 설치된 압력 센서(PS1; Pressure Switch)가 OFF인 상태
- 충전 준비단계: 탱크로리에서 헬륨이 배관에 공급되어 밸브 전단에 설치된 압력 센서(PS1)가 ON인 상태

- 충전 진행단계: 밸브가 개방되어 밸브와 헬륨 탱크 사이에 설치된 압력센서(PS2)가 ON인 상태

헬륨 충전을 위한 PLC 프로그램은 다음과 같이 메인 알고리즘과 시뮬레이션 알고리즘으로 구분되고 두 알고리즘은 동시에 운용된다.

메인 알고리즘은 Fig. 4와 같으며, 알고리즘의 주요 입력과 출력은 다음과 같다.

- 입력: 충전시작 버튼, 압력센서 PS1, PS2, 밸브 제어명령(개방/차단)
- 출력: 밸브 상태정보(개방/차단), 정상 충전 메시지, 비정상 충전 메시지(MSG)

메인 알고리즘은 다음과 같이 동작한다.

- “충전시작” 버튼을 누르고 3초 후 밸브 전

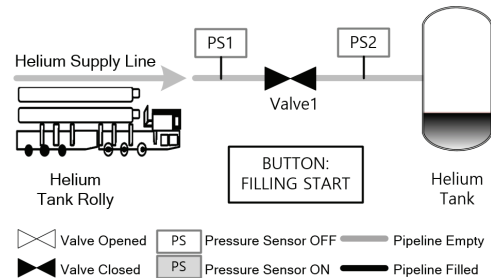


Fig. 1 Filling initial stage.

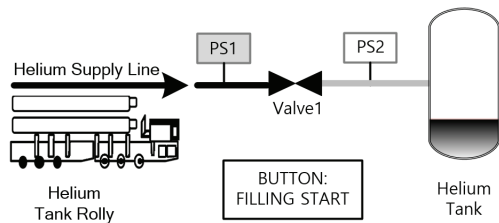


Fig. 2 Filling ready stage.

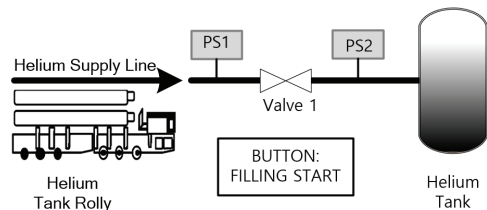


Fig. 3 Filling progress stage.

- 단 압력센서 PS1의 상태를 확인한다.
- 만약 PS1이 OFF인 충전 초기단계이면 “충전 시작” 버튼의 대기 상태로 분기하며,
- PS1이 ON인 충전 준비단계이면 밸브를 구동하여 헬륨이 공급되도록 한다.
- 밸브가 열리고 5초 후 압력센서 PS2가 ON인 충전 진행단계이면 “정상 충전”이라는 메시지를 모니터 상에 출력하고, 300초 후 밸브를 닫고 종료한다.
- 만일 PS2가 OFF로 충전 진행단계가 아니면 “비정상 충전”이라는 메시지를 출력 후 밸브를 닫고 종료한다.

시물레이션 알고리즘은 Fig. 5와 같고 다음과 같이 동작한다.

- 메인 알고리즘 충전시작 버튼이 눌리면 시물레이션 알고리즘은 이를 감지하고, 먼저 PS1을 ON으로 설정한다.
- 메인 알고리즘은 PS1이 ON인 경우이므로 Valve1을 개방한다.
- 시물레이션 알고리즘은 Valve1의 개방 상태를 확인 후 PS2를 ON으로 설정한다.
- 메인 알고리즘은 “정상 충전 “메시지 출력 후 300초 후에 밸브를 닫는다.

메인 알고리즘은 시물레이션 알고리즘 설정

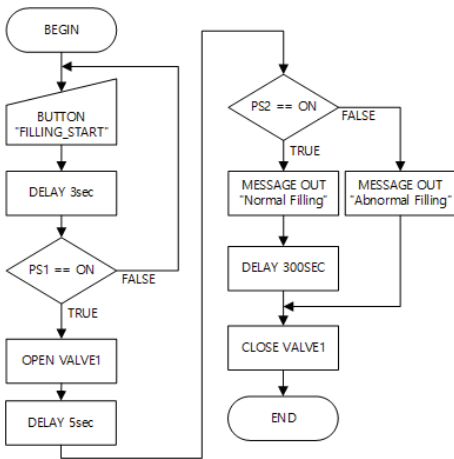


Fig. 4 Main algorithm.

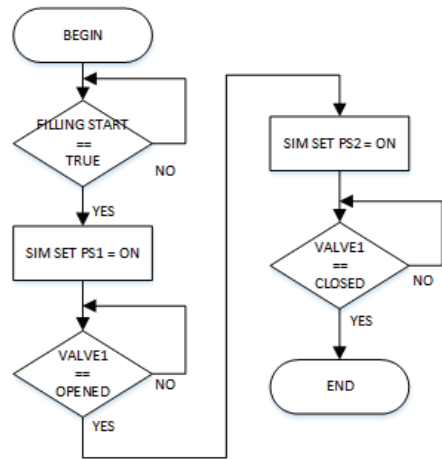


Fig. 5 Simulation algorithm.

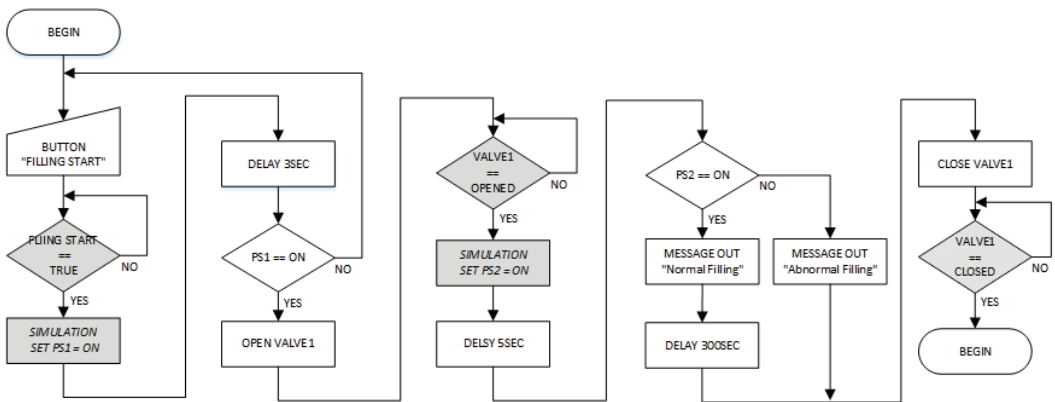


Fig. 6 Operation algorithm of PLC program.

값에 반응하여 동작하므로, 메인 알고리즘 구동 시 시뮬레이션 알고리즘의 설정 값을 변경하면 메인 알고리즘의 흐름을 유도하고 검증할 수 있다.

발사관제시스템에 구현된 PLC 프로그램들은 Fig. 6과 같이 메인 알고리즘과 시뮬레이션 알고리즘이 혼합된 형태로 구현된다. 별도의 시뮬레이션 장비 없이 운용자 콘솔에서 시뮬레이션 값 입력을 통한 알고리즘 검증은 편리성과 오류의 즉각적인 검출에 용이하나 장시간의 반복적 검증시험 과정과 작업자의 피로 누적 등으로 다양한 작업 오류가 발생된다. 이러한 문제를 해결하기 위해 스크립트 기반의 시험 절차서 작성과 스크립트 시뮬레이션 명령에 따른 예측 값을 라인 단위로 검증하는 시뮬레이션 입출력 자동화 검증 방법을 3.2절에서 제안한다.

3.2 제안 및 구현 방안

Fig. 7의 알고리즘 검증은 알고리즘 흐름을 좌우하는 압력센서 PS1과 PS2의 상태 정보를 강제 ON/OFF 모사 후 알고리즘 진행 상태를 확인하는 방법이다. 주요 시험 케이스는 압력센서 PS의 판단문의 조건에 따라 4개의 경우로 구분되며, Table 1의 시험 조건과 절차로 정리된다.

Table 1에 작성한 시험 절차서는 시험 절차서 편집기를 이용해 Fig. 8과 같은 스크립트 기반의 시험절차서로 변환한다. Fig. 8은 Table 1의 시험 항목 4번을 스크립트 기반의 시험 절차서로 기술한 것이다.

Fig. 8과 같이 작성된 스크립트 시험 절차서는 다음과 같은 명령으로 라인 단위의 명령과 검증이 자동 수행된다.

- START/END 명령(2, 12번 라인): 시험의 시작과 종료 시간을 기록하는 명령

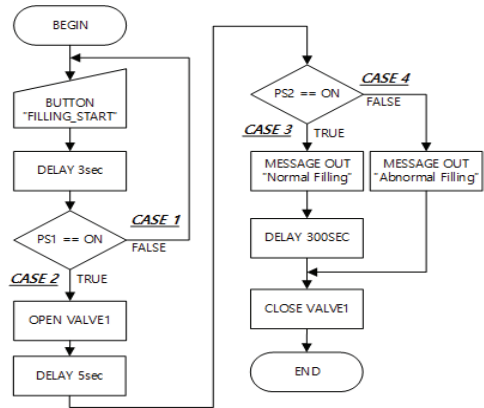


Fig. 7 Main algorithm of helium filling.

Table 1. Test conditions and procedures according to pressure sensors PS1, PS2 inputs.

No.	Test case	Test condition and procedure	No.	Test case	Test condition and procedure
1	CASE 1	Test condition: - PS1 == ON Test procedure: - Press button "Filling Start" - Simulate "PS1 ON" - Valve status check "OPEN"	2	CASE 2	Test condition: - PS1 == OFF Test procedure: - Press button "Filling Start" - Simulate "PS1 OFF " - Valve status check "CLOSE"
3	CASE 3	Test condition: - PS1 == ON, PS2 == ON Test procedure: - Press button "Filling Start" - Simulate "PS1 ON" - Valve status check "OPEN" - Simulate "PS2 ON " - Message check "Normal Filling" - Valve status check "CLOSE"	4	CASE 4	Test condition: - PS1 == ON, PS2 == OFF Test procedure: - Press button "Filling Start" - Simulate "PS1 ON" - Valve status check "OPEN" - Simulate "PS2 OFF" - Message check "Abnormal Filling" - Valve status check "CLOSE"

- PPULSE\* 명령(3번 라인): “충전시작”을 알리는 변수의 상태를 ON으로 1초 활성화 후 OFF로 변경하는 명령 \* Positive Pulse
- SLFON/SLFOFF\*\* 명령(4, 9번 라인): 디스크리트 센서 신호를 강제 ON/OFF 상태로 만듦.  
\*\* Signal Locking Force ON/OFF
- VERIFY 명령(5, 7, 10, 11번 라인): 센서, 밸브 또는 내부 메시지(MSG) 변수의 상태 값이 ON/OFF, OPEN/CLOSE, “정상/비정상 충전” 상태인지를 검증해 주는 명령
- TIMEDELAY 명령(6, 8번 라인): 초 단위의 대기 명령

중요한 점은 라인 단위로 명령과 검증이 수행되는 과정에 제어기에 탑재된 프로그램도 같이 동작하게 되며, VERIFY 명령이 사용된 라인에서 PASS or FAIL을 판별하게 된다. VERIFY 라인에서 FAIL이 발생하게 되면 시험 절차서는 대기 상태로 머무르게 되고, 개발자는 FAIL의 원인을 찾는 디버깅 과정을 수행해야 한다. 디버깅을 통해 프로그램 수정이 완료되면 기 작성된 시험 절차를 반복 수행한다. 검증은 이러한 과정을 반복적으로 수행하며 절차서의 마지막 라인까지 수행이 완료되면 검증이 완료된 것으로 판단한다.

INDEX	BREAK	COMMAND SCRIPT	TIMER	PASS/FAIL
1	<input type="checkbox"/>	### TEST CASE 4 ###		
2	<input type="checkbox"/>	START TEST //SAVE TIME OF START		
3	<input type="checkbox"/>	PPULSE ST_FILLING //ACTIVATION OF FILLING START		
4	<input type="checkbox"/>	SLFON PS1 //PS1 SENSOR ON BY FORCE		
5	<input type="checkbox"/>	VERIFY PS1 ON //VERIFY PS1 SENSOR ON		
6	<input type="checkbox"/>	TIMEDELAY 1 //ONE SECOND TIME DELAY		
7	<input type="checkbox"/>	VERIFY VALVE OPEN //VERIFY VALVE OPEN STATE		
8	<input type="checkbox"/>	TIMEDELAY 1 //ONE SECOND TIME DELAY		
9	<input type="checkbox"/>	SLFOFF PS2 //PS2 SENSOR OFF BY FORCE		
10	<input type="checkbox"/>	VERIFY PS2 OFF //VERIFY PS2 SENSOR OFF		
11	<input type="checkbox"/>	VERIFY MSG"ABNORMAL FILLING" //VERIFY MSG STATE		
12	<input type="checkbox"/>	END TEST //SAVE TIME OF END		
13	<input type="checkbox"/>	### TEST CASE 4 COMPLETE ###		

Fig. 8 Automated test sequence.

#### 4. 자동화 검증시스템을 이용한 시험

본 장에서는 제어알고리즘 검증 대상인 한국형발사체 발사대 고압가스공급 설비와 전자식 유공압 설비의 구성, 기능, 동작 개념을 설명하고 검증시스템을 이용한 시험 결과를 제시한다.

##### 4.1 검증 대상

한국형발사체 발사대 고압가스 공급설비는 총 16개의 자동 패널과 45개의 수동 패널로 구성되며, 저장용 가스 용기로부터 고압의 공기, 질소,

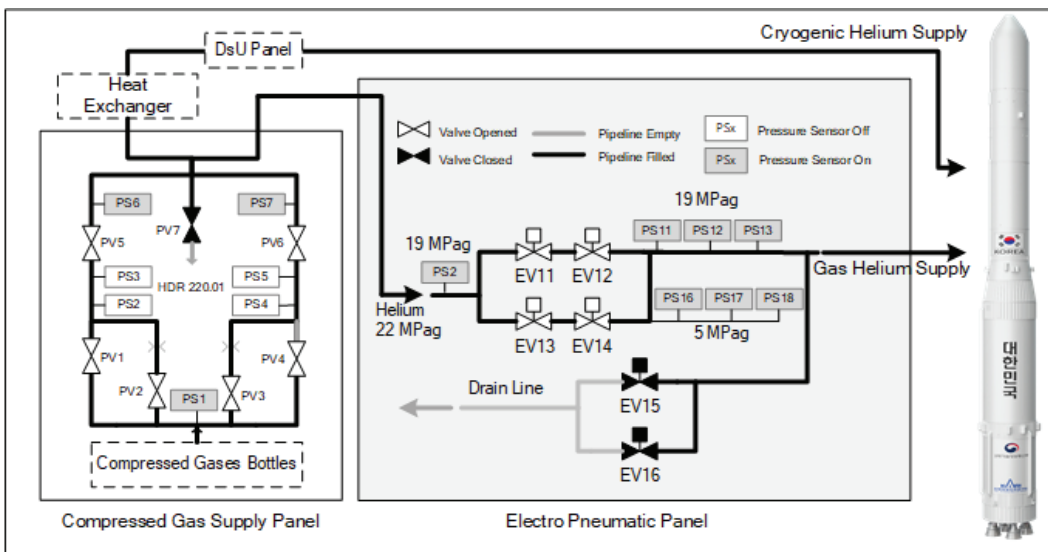


Fig. 9 Flow diagram of helium filling to KSLV-II.

헬륨 가스를 각 시스템별 사용 압력으로 감압 후 발사대 설비와 우주발사체에 공급하는 기능을 담당한다.

한국형발사체 발사대 전자식 유공압 설비는 총 12개 자동 패널로 구성되며, 고압가스 공급설비로부터 감압된 가스를 공급받아 발사체의 밸브 구동, 배관/엔진/밸브 퍼지, 탱크 가압, 헬륨 충전 등의 용도로 사용을 위한 공급 기능을 담당한다.

Fig. 9는 초저온 헬륨(Cryogenic Helium)과 고압의 상온 헬륨(Gas Helium)을 22 MPa의 압력으로 발사체에 공급하는 고압가스 공급패널(Compressed Gases Supply Panel)과 전자식 유공압패널(Electro Pneumatic Panel)의 연계 구조와 동작 개념의 예를 보여준다.

고압가스 공급 패널은 배관과 밸브 그리고 압력센서가 좌·우 이중화 구조로 구성되어 있다. 패널 입구에 설치된 PS1은 고압의 압력 공급의 유무를 감지하며, 출구에 설치된 PS6, PS7은 감압된 압력의 공급 상태를 감지한다. PS2~PS5는 감압된 압력이 예상보다 높을 때 안전밸브가 작동하여 감지되는 안전센서로 좌측(PV1, 2, 5) 또는 우측(PV3, 4, 6) 배관의 밸브를 긴급히 닫기 위한 용도로 사용된다.

고압의 상온 헬륨은 액체질소가 충전된 열교환기(Heat Exchanger)를 통해 90~120 K의 저온 헬륨으로 전환되며, 최종적으로 DSU(Delivery Supply Unit) 패널의 밸브를 개방하여 발사체에 초저온 헬륨 탱크 충전을 수행한다.

고압가스 공급 패널로부터 헬륨 가스를 공급받는 전자식 유공압 패널은 배관 및 밸브가 병렬식 이중화 구조로 되어 있고, 밸브의 전단과 후단의 압력을 감지하는 센서(PS)가 부착되어 있다. 패널은 가스 공급과 중지 명령에 따라 EV11~EV14 밸브를 개·폐하여 고압의 헬륨가스를 공급 또는 중지하는 기능을 수행하며, 해압 명령 수신 후 EV15, EV16 밸브를 개방하여 가스헬륨 배출을 수행한다.

대부분의 고압가스 공급 패널과 전자식 유공압 패널은 Fig. 9에서 제시된 구성, 기능 그리고 동작 개념이 유사하다.

## 4.2 시험 결과 및 분석

본 절에서는 고압가스 공급설비와 전자식 유공압패널설비에 대한 제어알고리즘 검증 시험결과를 제시한다.

두 설비에 대한 스크립트 시험절차서는 총 268개로 구성하고, 자동화 검증시스템을 적용하여 제어시스템에 탑재되어 있는 알고리즘 검증을 공장수락시험(FAT: Factory Acceptance Test)과 현장수락시험(SAT: Site Acceptance Test) 기간에 진행하였다. 검증시험은 각 패널별로 다음의 항목을 측정하여 결과에 대한 비교/분석을 수행하였다.

- 전체 시험시간(TMT: Test Measurement Time)
- 시험절차서 구동시간(RTTS: Run Time of Test Sequence)
- 작업자 조작시간(OMT: Operator Manipulation Time)
- 전체 시험시간(TMT) = 시험절차서 구동시간(RTTS) + 작업자 조작시간(OMT)

Table 2는 고압가스 공급 패널에 대한 시험 측정 결과이다.

고압가스 공급 패널은 제어알고리즘 검증에 패널 1개당 11~14개의 스크립트 시험 절차서로 구성된다. 패널 전체 수량에 사용된 시험절차서는 총 182개이다.

공장수락시험(FAT)에서 측정한 고압가스 공급 패널 1개당 TMT는 평균 34분으로 측정되었다. TMT 시간은 제어알고리즘 검증의 시작 선언부터 종료 선언까지 걸린 시간을 나타낸다. 이 시간은 작업자의 시험 준비시간, 시험절차서 구동 시간(RTTS), 시험 결과저장과 예측 불가능 알고리즘 디버깅 등의 시간들이 포함되어 있다.

ADR8C1, HDR65C01, NDR25C01의 TMT 시간이 다른 패널에 비해 크다. 그 이유는 알고리즘 오류에 따른 디버깅 시간이 추가되었기 때문이다.

알고리즘 오류가 없는 경우 프로그램 개발자가 시험 절차서에 따라 시뮬레이션을 설정하고, 결과 값과 예측 값을 비교 검증하는 시험절차를

수작업으로 수행하는데 패널 1개당 약 60분 정도의 RTTS 시간이 소요되며, 이는 작업자의 숙련도에 따라 증가 또는 감소될 수 있다. 하지만, Table 2에 제시된 RTTS는 평균 약 12분으로 스크립트 시험절차서와 자동화 검증시스템을 활용하여 패널 1개당 약 48분의 단축 효과를 보였다.

OMT는 디버깅 등을 포함한 작업자의 매뉴얼 조작이 얼마나 있었는지를 보여준다. 공장수락시험(FAT)에서 OMT 시간이 불규칙하게 나타나는 이유는 개발자의 초기 검증시험에 대한 시행착오가 반영된 것이다.

현장수락시험(SAT)에서는 FAT에서 1차 검증이 완료된 알고리즘을 재시험 하는 경우로 패널 TMT 시간이 평균 약 34분에서 20분으로 OMT가 21분에서 8분으로 감소됨을 확인하였다. TMT나 OMT의 감소 원인은 작업자의 숙련도 상승과 알

고리즘 오류로 인한 디버깅 시간 등의 요인이 제외되었기 때문으로 볼 수 있다. RTTS는 FAT에서 사용된 스크립트 시험절차서를 동일하게 SAT에 적용하여 큰 차이가 없음을 확인하였다.

Table 3은 전자식 유공압 패널에 대한 시험 측정 결과이다.

전자식 유공압 패널은 제어 알고리즘 검증에 패널 1개당 6~8개의 시험 항목과 스크립트 시험절차서로 구성되었으며, 패널 전체 수량에 사용된 시험절차서는 총 86개이다.

전자식 유공압 패널의 Type은 5가지 유형으로 Table 3과 같이 구분된다.

전자식 유공압 패널 시험 역시 TMT는 전체적으로 SAT보다 FAT에서 큰 값을 보인다. 그 이유는 1차 검증 과정인 FAT에서 알고리즘 오류와 운용 화면의 수정사항이 다수 발생하였고, 작업자의 시험 경험 부족과 초기 시험에 따른 시행착오가 역시 많이 발생했기 때문이다.

Table 2. The test result of compressed gases supply panel control algorithms.

No.	Panel name	Num. of test case	FAT / SAT		
			TMT* (min.)	RTTS** (min.)	OMT (min.)
1	ADR*8C01	14	70/32	11/12	59/20
2	ADR6C23	11	35/23	15/13	20/10
3	ADR47C22	11	25/21	13/12	12/9
4	ADR40C21	11	30/20	12/12	18/8
5	NDR**4C24	11	24/20	12/12	12/8
6	NDR4C25	11	26/18	12/12	14/6
7	ADR10C01	11	21/18	9/9	12/9
8	ADR150C01	11	22/19	12/12	10/7
9	HDR***30C01	11	20/17	12/12	8/5
10	HDR65C01	11	53/17	12/12	41/5
11	HDR220C01	11	29/18	12/12	17/6
12	NDR4C02	11	29/17	13/11	16/6
13	NDR10C01	11	20/17	12/11	8/6
14	NDR25C01	11	80/17	12/12	68/5
15	NDR65C01	11	19/17	12/12	7/5
16	NDR4C01	14	35/35	16/17	19/18
	Sum	182	538/326	197/193	341/133
	≈ Average rack test time		≈ 34/20	≈ 12/12	≈ 21/8

\*TMT: Total Test Time, \*\*RTTS: Run Time of Test Sequence  
 \*\*\*OMT: Operator Manipulation Time  
 \*ADR: Air Delivery Rack, \*\*NDR: Nitrogen Delivery Rack  
 \*\*\*HDR: Helium Delivery Rack

Table 3. The test result of site electro pneumatic panel control algorithms.

Type	Panel name	Num. of test case	FAT / SAT		
			TMT* (min.)	RTTS** (min.)	OMT (min.)
1	1EPP*1	8	353/169	50/53	303/116
	1EPP3	8	294/62	62/55	232/7
	2EPP1	8	110/60	50/51	60/9
	3EPP1	8	134/58	55/50	79/8
2	1EPP2	6	137/131	44/44	93/87
	2EPP2	6	57/48	41/41	16/7
	2EPP4	6	236/48	38/41	198/7
	3EPP3	6	74/44	41/40	33/4
3	1EPP4	8	365/78	78/49	287/29
	1EPP5	8	104/58	55/47	49/11
4	2EPP3	6	57/33	28/28	29/5
5	3EPP2	8	191/60	50/51	141/9
	Sum	86	2112/849	592/550	1520/299
	≈ Average rack test time		≈ 176/71	≈ 49/46	≈ 127/25

\*TMT: Time of Test Measurement  
 \*\*RTTS: Run Time of Test Sequence  
 \*\*\*OMT: Operator Manipulation Time  
 \*EPP: Electro Pneumatic Panel



RTTS는 몇 개의 패널을 제외한 대부분의 패널이 FAT나 SAT에서 모두 유사한 측정 시간을 보이며, 1EPP3, 1EPP4, 1EPP5 등은 약간의 시간 차이를 보인다. 이는 디버깅을 위해 스크립트 시험절차서가 구동 중 일시 정지의 상태에 머물러 있었기 때문이다.

프로그램 개발자의 수작업에 의한 RTTS는 패널 1개당 약 150분이 소요된다. 본 논문에서 스크립트 시험절차서와 자동화 검증시스템을 활용한 RTTS는 평균 49(FAT)과 46(SAT) 분으로 패널 1개당 약 100분의 단축 효과를 보였다.

OMT의 경우 SAT에서 많은 시간 절약을 보인다. 그 이유는 역시 FAT를 통해 시험 준비와 방법에 대한 경험 습득과 작업자의 숙련도 그리고 알고리즘 동작에 대한 이해도가 축적되어 작업자의 매뉴얼 조작이 많이 개선되었고, 알고리즘에 오류가 많이 제거되었기 때문이다.

본 논문의 시험 결과로 스크립트 시험절차서와 자동화 검증시스템을 활용하여 작업 오류와 시험절차서 구동시간(RTTS)을 단축하였고, 결과적으로 전체 시험시간(TMT)을 효과적으로 단축함을 확인하였다. 만약, 자동화 시험절차서 대신 수작업에 기반 한 시험이 수행되었다면 작업자의 컨디션 또는 작업오류에 따라 RTTS가 매우 불규칙하게 변동되어 TMT의 지연 요인으로 작용되었을 것이다.

## 5. 결 론

본 논문에서는 체계적인 시험절차와 방법을 규정하고 작업오류를 최소화하며 검증시간을 단축하기 위한 스크립트 기반의 자동화 검증시스템을 구현하고 실제 시스템에 적용하여 검증 시간이 효과적으로 단축됨을 보였다. 그러나 시험 계획 시 수작업에 의존하는 스크립트 시험절차서 작성은 비록 사전 작업이기는 하지만 경우의 수가 많은 복잡한 알고리즘의 경우 많은 시간이 소요될 수 있는 문제를 지니고 있다. 이에 향후 연구과제로 개발 초기 작성되는 Flow-Chart에서 원하는 시험 케이스를 구성하고 선택하여 스크

립트를 자동으로 변환해 주는 생성기를 개발하여 검증시스템에 추가하는 연구를 수행하고자 한다.

## References

1. An, J.C., Moon, K.R. and Oh, I.S., "Verification of Control System Algorithms for Compressed Gas Supply Facility in Launch Complex," *Journal of the Korean Society of Propulsion Engineers*, Vol. 23, No. 2, pp. 95-103, 2019.
2. Korea Atomic Energy Research Institute (KAERI), "State-of-the-Art Report for the testing and formal verification methods for FBD program," AR-901, 2011.
3. Jung, M.P. and Kim, K.N., "The Design and Implement of Program Simulator for testing PLC-based control systems," *In Proceedings of KIISE Conference*, Seoul, Korea, pp. 63-65, Oct. 1999.
4. An, J.C., Moon, K.R. and Oh, I.S., "Design and Implementation of Simulator of Launch Control System," *Journal of the Korean Society for Aeronautical & Space Sciences*, Vol. 44, No. 8, pp. 657-665, 2016.
5. Jung, M.J., Shin, Y.G., Choi, S.W., Moon, G.W., Seo, H.J., Lee, S.J. and Cho, H.J., "Application of a realtime plant simulator to develop a PLC for a space thermal environment chamber," *In Proceedings of KSAS Conference*, Korea, pp. 172-175, Nov. 2003.
6. Suh, J.K., Lee, M.S. and Hong, J.H., "Digital Control System Validation using the Simulator Models for 500MW Standard Type Fossil Power Plant," *Journal of the Korea Society for Simulation*, Vol. 19, No. 3, pp. 71-79, 2010.
7. Bun S.H. and Whang D.H., "Development

- of Power Plant Simulator for Control System Verification & Validation," *Journal of the Korea Society for Simulation*, Vol. 19, No. 1, pp. 41-51, 2010.
8. Bae, S.H. and Kim, Y.M., "A Case Study of Virtual PLC Validation System's Implementation: In Case of An Automobile Trim Line," *Journal of the Korea Society for Simulation*, Vol. 19, No. 2, pp. 9-16, 2010.
  9. Choi, S.U., Park, S.C. and Wang, G.N., "Virtual Plant building Process for PLC Code Verification in Automotive body Assembly System," *In Proceeding of KIIIE Conference*, Seoul, Korea, pp. 803-808, Nov. 2008.
  10. Lime, C.G., Ahn, Y.S., Park, S.C. and Wang, G.N., "A Study of PLC Simulation Applying for Steel Product Line," *In Proceeding of CAD/CAM Conference*, Pyeongchang, Korea, pp. 120-124, Jan. 2011.
  11. Park, S.C., Park, C.M., Wang, G.N., Kwak, J. and Yeo, S., "PLCStudio: simulation based PLC code verification," *In Proceedings of the 40th Conference on Winter Simulation, Winter Simulation Conference*, Miami, FL, U.S.A., pp. 222-228, Dec. 2008.
  12. Yoo, J., Cha, S., and Jee, E. "A verification framework for FBD based software in nuclear power plants," *In Proceedings of the 2008 15th Asia-Pacific Software Engineering Conference*, Beijing, China, pp. 385-392, Dec. 2008.
  13. Schlich, B., Brauer, J., Wernerus, J. and Kowalewski, S. "Direct model checking of PLC programs in IL," *IFAC Proceedings*, Bari, Italy, Vol. 42, No. 5, pp. 28-33, Jun. 2009.
  14. Kwon, M.H. and Shin, S.C., "Automatic Translations for Model Checking of LD Programs," *Journal of Korean Institute of Information Scientists and Engineers*, Vol. 16, No. 2, pp. 201-206, 2010.
  15. Canet, G., Couffin, S., Lesage, J.J., Petit, A., and Schnoebelen, P., "Towards the automatic verification of PLC programs written in Instruction List," *In 2000 IEEE International Conference on Systems, Man and Cybernetics 'Cybernetics Evolving to Systems, Humans, Organizations, and Their Complex Interactions'*, Nashville, TN, USA, Vol. 4, pp. 2449-2454, Oct. 2000.
  16. Pavlovic, O. and Ehrich, H.D., "Model checking PLC software written in function block diagram," *In Proceedings of the 2010 Third International Conference on Software Testing, Verification and Validation*, Paris, France, pp. 439-448. Apr. 2010.
  17. Soliman, D., Thramboulidis, K. and Frey, G. "Transformation of function block diagrams to Uppaal timed automata for the verification of safety applications," *Journal of Annual Reviews in Control*, Vol. 36, No. 2, 338-345, Dec. 2012.
  18. Adiego, B.F., Darvas, D., Vinuela, E.B., Tournier, J.C., Suárez, V.M.G. and Blech, J.O., "Modelling and Formal Verification of Timing Aspects in Large PLC Programs," *IFAC Proceedings*, Cape Town, South Africa, Vol. 47, No. 3, pp. 3333-3339, 2014.
  19. Park, C.M., "Formal Verification of PLC Program Safety in Manufacturing Automation System," *Journal of Korea Safety Management and Science*, Vol. 17, No. 1, pp. 179-192, 2015.
  20. Zhou, M., He, F., Gu, M. and Song, X., "Translation-Based Model Checking for PLC Programs," *33rd Annual IEEE International Computer Software and Applications Conference*, Seattle, WA, U.S.A., pp. 553-562, Jul. 2009.
  21. Soliman D. and Frey, G., "Verification and validation of safety applications based on PLCopen safety function blocks," *Journal of*

- 
- Control Engineering Practice*, Vol. 19, No. 9, pp. 929-946, 2011.
22. Jee, E.K., Jeon, S.J. and Cha, S.D., "A Formal Verification Technique for PLC Programs Implemented with Function Block Diagrams," *Journal of Korean Institute of Information Scientists and Engineers*, Vol. 15, No. 3, pp. 211-215, 2009.
23. Xion, J., Zhu, G., Huang, Y. and Shi, J. "A User-Friendly Verification Approach for IEC 61131-3 PLC Programs," *Journal of Electronics* 9, No. 4, pp. 572, 2020.