

안전등급 PLC용 C-코드 발생기 시험

김 석 주, 천 중 민, 이 중 무, 권 순 만
한국전기연구원 계속제어그룹

Test of a C-code generator for safety-grade PLCs

Seog-Joo Kim, Jong-Min Cheon, Jong-Moo Lee, Soonman Kwon
I&C Group, Korea Electrotechnology Research Institute (KERI)

Abstract - 본 논문에서는 안전등급 PLC(Programable Logic Controller) LD/FBD(Ladder/Function Block Diagram) 프로그램을 C-코드로 변환하는 C-코드 발생기의 시험에 관해서 다룬다. 개발된 C-코드 발생기는 로직 분석, 오류 검출, C-코드 생성 등 각 부분 별로 시험 항목을 설정하고 검증하는 방법에 대해서 서술한다. 또한 C-언어로 작성된 사용자 정의 함수/함수 블록에 관한 시험 항목을 기술한다.

그럼 개발을 위한 편집, 다운로드, 실행 및 모니터링 등 통합개발환경(IDE; Integrated Development Environment)을 제공한다.

PLC와 PC는 그림 2와 같이 RS-232C로 연결되며 그림 2에서 좌측이 pSET이 운용되는 컴퓨터이고 우측이 개발된 프로그램이 다운로드 되는 안전등급 PLC이다.

1. 서 론

안전등급 PLC(Programable Logic Controller)란 안전을 최우선(safety critical)으로 하는 플랜트의 제어 및 감시에 사용될 목적으로 개발되고 제작된 PLC를 말한다. 안전 최우선 플랜트의 대표적인 예가 원자력발전소, 고속철도, 항공기 등일 것이다. 이들 플랜트를 제어하거나 감시하는데 사용되는 PLC는 일반적인 플랜트에 사용되는 PLC와는 하드웨어 및 소프트웨어에서 전혀 다른 개발 과정과 제작 과정을 거치게 된다. 즉, 철저한 검증과 시험을 바탕으로 개발하거나 제작한다.

일반적으로 PLC 응용 프로그램은 PLC 국제 표준 언어인 IEC 61131-3 언어로 개발되고 이것을 컴파일 과정을 거쳐서 타겟 PLC에 맞는 기계어로 변환되고 실행파일을 PLC에 다운로드하여 실행시키게 된다[1]. 이 과정에서 안전을 위한 가장 중요한 부분은 IEC 61131-3 언어를 기계어로 바꾸는 과정인 컴파일 과정이다. IEC 61131-3 언어는 그래픽 언어로 엔지니어의 입장에서 보면 누구나 쉽게 확인 및 검증을 수행할 수 있다. 하지만 이 IEC 61131-3 언어가 타겟 PLC에 맞는 코드로 올바르게 변환되었는지 여부는 많은 논란의 소지가 있으며 PLC 소프트웨어의 확인/검증 과정의 핵심이다.

“원전 계속제어 시스템 개발 사업”(이하 KNICS; Korea Nuclear I&C System)에서 개발된 안전등급 PLC는 컴파일 과정에 확인/검증이 용이하게 하기 위해서 IEC 61131-3 언어인 LD/FBD 프로그램을 중간 단계에서 C-소스 프로그램으로 변환하는 C-code 발생기를 개발하였다. 이것은 C-코드로 변환된 프로그램은 확인/검증이 용이할 뿐만 아니라 C-코드를 컴파일하기 위한 입증된 COTS (Commercial-Off-The-Shelf) 컴파일러를 사용할 수 있기 때문이다.

본 논문에서는 개발된 C-code 발생기의 소프트웨어 확인/검증의 한 과정인 시험에 관해서 기술한다[3-5]

2. 안전등급 PLC 프로그램 개발 환경

2.1 안전등급 PLC 개요

KNICS에서 개발된 안전등급 PLC는 원자력발전소 안전 관련 계통에 적용하기 위해서 개발된 PLC로 POSAFE-Q라고 부르며 그림 1과 같이 메인 채시(main chassis), 전원모듈(PWR), 프로세서모듈(PM), 통신모듈(CM), 디지털 입출력 모듈(DIO), 아날로그 입출력 모듈(AIO) 그리고 펄스 카운터 모듈(PCM) 등의 모듈로 구성된다.

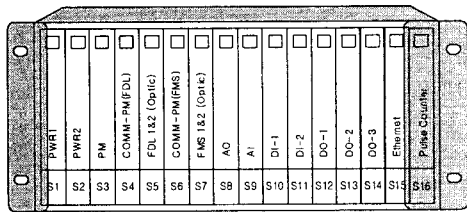


그림 1. POSAFE-Q의 메인 채시 및 모듈 구성

POSAFE-Q의 메인 채시는 19인치 표준 규격을 만족하며 응용에 따라 여러 개의 입출력 모듈 및 통신 모듈을 조합하여 사용할 수 있다.

2.2 안전등급 제어기용 프로그램 개발 도구 개요

KNICS PLC의 프로그램 개발 도구는 pSET (POSAFE-Q Software Engineering Tool)이라고 부르며 안전등급 PLC 인 POSAFE-Q PLC 모

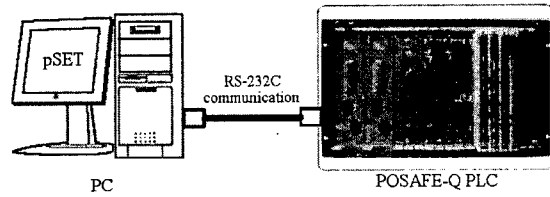


그림 2. pSET의 개발 환경

pSET은 마이크로소프트의 윈도우즈 환경에서 운용되며 사용자 인터페이스를 포함한 IDE 화면은 그림 3과 같다.

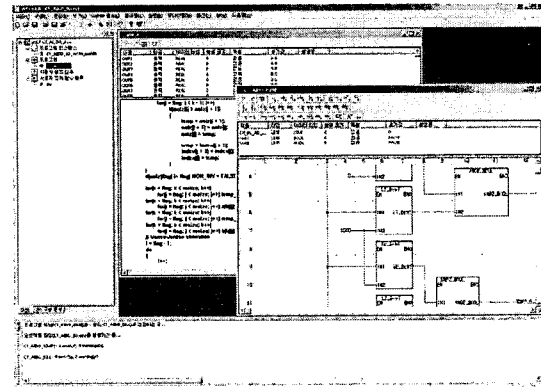


그림 3. pSET IDE 환경

다음은 pSET의 주요 특징은 다음과 같다.

- 하나의 구성(configuration)에서 사용하는 I/O, 태스크 및 태스크로 실행되기 위한 프로그램들을 디렉토리 형식으로 계층적으로 관리한다.
- 단일 IDE를 통하여 통합적으로 구성을 관리할 수 있는 환경을 제공한다.
- IEC 61131-3의 규약에 따른 프로그램 언어 LD/FBD를 지원한다.
- 인터프리터 방식이 아닌 컴파일 방식으로 실행속도가 빠르다.
- LD/FBD로 작성한 프로그램을 C-코드로 변환한 후에 상용 컴파일러를 통하여 최종 기계어 코드를 생성하기 때문에 다음과 같은 장점이 있다.
 - 타겟 PLC의 CPU가 업그레이드 또는 변경되어도 상용 C-컴파일러만 변경하면 되므로 향후 업그레이드가 용이하다.
 - 하드웨어 PLC가 아닌 PC를 이용하는 Soft-PLC로의 확장이 용이하다.
 - 상용 컴파일러의 검증된 코드 최적화 기능 등을 이용할 수 있기 때문에 최적화된 기계어 코드 생성이 가능하다.
 - 개발 과정에서 사용자가 작성한 프로그램이 올바르게 C-코드로 변환되었는지 확인/검증이 용이하다.
- IEC 61131-3의 규약에 따른 프로그램 언어 이외에 ANSI-C 코드로 작성한 프로그램을 사용자 정의 함수 또는 함수블록 형태로 추가하여 사용하는 것이 가능하다.
- 시뮬레이션 모듈을 지원한다. 즉, PLC로 다운로드하기 전에 PC에서 시뮬레이션을 통하여 동작을 미리 점검하는 것이 가능하다.

- pSET에서 편집환경과 통합된 모니터링 환경을 지원한다.
- 변수 강제 설정(force on/off/reset)을 지원한다.
- 링(rung) 단위로 중단점을 설정하고 단일 스텝으로 실행할 수 있는 디버깅 모드를 지원한다.
- 프로그램에 사용된 변수(tag)의 가져오기/내보내기 기능이 있다.
- 보안 및 형상 관리 기능을 지원한다.

위 기능 중에서 안전등급 PLC 요건상 가장 중요한 부분은 IEC 61131-3 언어가 기계어로 바뀌는 부분인 컴파일러일 것이다.

2.3 C-코드 발생기

KNICS 사업에서 개발된 C-코드 발생기는 PLC의 프로그램 단위인 링(rung)을 노드(node)와 브랜치(branch)로 나누어 추출하고 추출된 노드와 브랜치를 C-코드로 변환하는 과정을 반복하게 된다. 다음에 pSET에서 LD/FBD 프로그램이 C-코드로 변환되는 과정을 간단하게 설명한다.

- 노드 : 셋 이상의 브랜치들이 만나는 부분 또는 하나의 FBD에 속한 원소의 집합
 - 브랜치 : 노드와 노드 사이를 연결하는 일렬 형태의 원소들의 배열
- 노드와 브랜치의 집합은 노드와 브랜치 사이의 연결 관계에 따라 다음과 같이 각 노드를 기준으로 병렬로 연결된 브랜치, 직렬로 연결된 브랜치 등을 하나의 브랜치로 병합하고 여러 개의 브랜치로 나누어 지는 노드는 여러 개의 브랜치로 분리하는 과정을 반복하여 더 이상 병합 또는 분리할 수 없는 단일 브랜치로 변환될 때 까지 변환을 반복한다.

3. C-코드 발생기 시험

C-코드 발생기 소프트웨어는 크게 세 개의 컴포넌트로 분류할 수 있다. 즉, C-코드 생성 컴포넌트, 기계어 프로그램 생성 컴포넌트 그리고 C-언어 사용자 정의 함수/함수블록을 처리하는 컴포넌트이다. 여기서의 주된 시험 대상은 LD/FBD 제어 언어로 작성된 프로그램 유닛을 분석하여 오류나 경고 사항들을 검출하여 사용자에게 알려 주거나 C-코드를 생성하는 기능을 가지는 컴포넌트이다. 표 1과 2에 C-코드 생성기능과 사용자 정의 함수/함수블록에 관한 주요 기능 시험 항목을 기술하였다.

표 1. 사용자 정의 함수/함수 블록의 시험 항목

시험 항목	설명
LoadCc	사용자가 작성한 C-언어 사용자 정의 함수/함수 블록을 처리하기 위하여 프로그램 유닛 파일을 불러오는 기능
SearchIncludedFBDList	사용자가 C-언어 사용자 정의 함수/함수 블록 작성 시에 사용한 FBD 리스트를 조사하는 기능
MakeUDFVarFile	C-언어 사용자 정의 함수 블록의 변수 파일을 생성하는 기능
MakeUDFHeaderFile	C-언어 사용자 정의 함수 블록의 헤더 파일을 생성하는 기능.
MakeInitFile	C-언어 사용자 정의 함수 블록의 초기화 파일을 생성하는 기능
MakeUDFCFile	C-언어 사용자 정의 함수 블록의 메인 소스 파일을 생성하는 기능
MakeUDFCHeaderFile	C-언어 사용자 정의 함수의 헤더 파일을 생성하는 기능
MakeUDFCCFile	C-언어 사용자 정의 함수의 메인 소스 파일을 생성하는 기능
MakeUDFCCFile	C-언어 사용자 정의 함수의 메인 소스 파일을 생성하는 기능

4. 결 론

본 논문에서는 KNICS에서 개발한 안전등급 PLC의 응용 프로그램 개발 도구인 pSET의 C-코드 발생기의 시험에 대해서 개략적으로 서술하였다. 이를 바탕으로 개발된 C-코드 발생기를 시험하였거나 또는 시험 중에 있다.

[참고 문헌]

- [1] IEC Standard 61131-3: Programmable Controllers - Part 3, IEC 61131, 1993.
- [2] 구경모외 4인, "Development of Application Programming Tool for Safety Grade PLC (POSAFE-Q)", 원자력학회 춘계학술대회, 2006.

- [3] 천종민, "pSET 컴파일러 컴포넌트 시험 계획서", 한국전기연구원, 2007.
- [4] 천종민, "pSET 컴파일러 컴포넌트 시험 절차서", 한국전기연구원, 2007.
- [5] 천종민, "pSET 컴파일러 컴포넌트 시험 결과 보고서", 한국전기연구원, 2007.

표 3. C-코드 발생기의 주요 시험 항목

시험 항목	설명
CheckConnection	원소들 각자에 요구되는 연결 방향을 파악하고 정상적으로 연결되었는지 검사하는 기능
MakeNode	LD/FBD 제어 언어 프로그램 유닛의 각 원소를 검사하여 노드인지 아닌지 그 여부를 판단하고 노드라면 일반 노드와 FBD 노드로 구분하여 그 배열을 만드는 기능
MakeBranch	LD/FBD 제어 언어 프로그램 유닛의 각 원소를 검사하여 브랜치 원소들을 추려내어 임의의 노드와 그 이웃 노드 사이에 일렬로 연결된 브랜치 원소들을 묶어서 하나의 브랜치를 만드는 기능
LinkNodesBranches	생성된 노드들과 브랜치들을 서로 연관 관계를 조사하고 서로 연결하는 기능
오류 검출 1	LD/FBD 제어 언어 프로그램 유닛 상의 원소들의 연결 관계를 조사하고 연결에 이상이 있으면(즉 연결되어야 하는 방향에 연결이 되어 있지 않으면) 오류라고 판단하는 기능
오류 검출 2	LD/FBD 제어 언어 프로그램 유닛 상의 원소들 중 문자열이 필요한 원소들의 문자열의 존재 여부를 조사하고 없는 경우에 오류라고 판단하는 기능
오류 메시지 생성	LD/FBD 제어 언어 프로그램 유닛의 컴파일 도중에 발견된 오류 및 경고 사항에 관하여 메시지를 작성하는 기능
GetRung	LD/FBD 제어 언어 프로그램 유닛의 각 원소들의 연결 관계를 추적하여 서로 완전히 연결되어 있는 하나의 링을 구성하는 브랜치 배열과 노드 배열을 추출하는 기능
BranchToCode	LD/FBD 제어 언어 프로그램 유닛의 한 링에 포함되어 있는 각각의 브랜치를 해당하는 로직의 C-코드로 변환하는 기능
브랜치 합병	LD/FBD 제어 언어 프로그램 유닛의 한 링에 포함되어 있는 C-코드로 변환된 브랜치들을 합병하여 최종적으로 하나의 링이 하나의 브랜치로 표현되도록 변환하는 기능
WriteToMainCode	LD/FBD 제어 언어 프로그램 유닛에서 하나의 브랜치로 합쳐진 링의 코드 변환을 마친 후에 링의 코드를 주 코드로 저장하는 기능
GenerateHeaderFile	LD/FBD 제어 언어 프로그램의 변수 및 로직 함수들을 정의하는 프로그램 헤더 파일을 생성하고 해당 프로그램과 연동된 태스크 관련 변수들과 프로그램 변수들을 초기화하는 초기화 소스 파일을 생성하는 기능
SaveCode	LD/FBD 제어 언어 프로그램 유닛(서브 프로그램)의 링의 개수를 정의하는 서브 프로그램 헤더 파일을 생성하고 서브 프로그램 로직을 C 언어로 번역한 C-코드를 포함하는 서브 프로그램 소스 파일을 생성하는 기능
MakeMainFile	LD/FBD 제어 언어 프로그램의 전체 메인 루틴으로서 프로그램에 소속된 서브 프로그램들을 함수 형식으로 호출하는 코드를 포함한다. 이 메인 프로그램소스 파일을 생성하는 기능