

IEC1131-3에 입각한 제어 시스템 설계 프로그램 개발

허우정, 신경봉*, 김응석*, 김문철*, 박정민**, 김성태***

*삼성종합기술원 산업전자연구실

**한국과학기술연구원 정보전자연구부

***삼성중공업 기전연구소

Development of Control System Design Program Based on IEC1131-3

Woojung Huh*, KyeongBong Shin*, Eung-Seok Kim*, MoonCheol Kim*, Jungmin Park**, SungTae Kim***

*System Control Laboratory, Samsung Advanced Institute of Technology

**Division of Electronics and Information Technology, Korea Institute of Science and Technology

***Research Institute of Machinery & Electrotechnology, Samsung Heavy Industries Co. Ltd.

Abstract - IEC1131-3 Specification of Programming Controller is established in 1994 and consists of 3 graphical languages and 2 textual languages. It is used in PLC and small scale controller because of its uniformity and extensibility. This paper describes Soft Logic Designer which is a graphical and textual programming editor for IEC1131-3 programming languages. Soft Logic Designer is developed with Object Orient Language, C++ under Microsoft Windows 95. It has two graphic editors for Sequential Function Chart and Function Block Diagram and one textual editor for Structured Text. Users can efficiently write high-level programs with mouse and menu buttons.

1. 서론

프로그래밍 콘트롤러 제어기는 고유의 프로그래밍 언어와 프로그래밍 툴을 이용하고 있다. PLC도 Ladder Diagram이라는 한가지 언어를 이용하고 있으나 화면에 나타나는 모양만 흡사할뿐 생신자가 다른면 오브젝트코드의 호환성은 말할 것도 없고 소스차원에서의 호환도 전혀 이루어지지 않고 있다. 이를 해결하고자 제시된 IEC1131-3 규격은 통일성 및 확장성을 인정받아 PLC뿐만아니라 여러 제어기의 표준화된 프로그래밍 언어로서 사용되고 있다.

본 논문에서는 IEC1131-3에서 정의된 5가지 언어중 상위 수준의 프로그래밍에 많이 이용되는 SFC(Sequential Function Chart), FBD(Function Block Diagram), ST(Structured Text)를 사용하여 그래픽 화면에서 메뉴 버튼을 이용하여 마우스만으로 프로그래밍 할수 있는 제어기 프로그래밍 툴을 제안한다.

2. 시스템의 구성

2.1 하드웨어의 구성

SLD 툴의 필요 시스템은 윈도우95가 실행가능한 486DX-2 66MHz이상의 CPU, 8MB이상의 시스템 메모리, 500MB이상의 하드디스크, 고해상도 컬러 그래픽을 표현할수 있는 SVGA급이상의 비디오 보드와 14인치 이상의 컬러 모니터를 갖춘 PC 호환기종과 SLD 툴로 작성된 프로그램을 실행하는 콘트롤러로 구성된다. 시스템 다이어그램은 다음과 같이 구성된다.

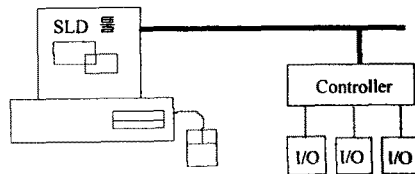


그림 1. H/W 시스템 구성도

2.2 소프트웨어의 구성

윈도우95 환경에서 마이크로소프트사의 윈도 응용 프로그램 개발 툴인 Visual C++ 4.0(이하 VC) 이용하여 작성하였다. VC는 클래스 라이브러리의 MFC 4.0을 지원하여 윈도우95 상에서 객체지향적인 프로그램의 작성에 적합하다. 프로그램의 유지 보수에 많은 장점이 있는 C++를 이용하여 작성하였다.

SFC, FBD, ST 편집기는 MFC를 이용하여 작성되었으며 객체지향 프로그램의 특징인 캡슐화, 추상화, 상속성을 이용하여 프로그래밍 되어 향후 업그레이드와 유지보수가 쉽도록 설계되었다. 전체적인 소프트웨어의 구조는 다음과 같다.

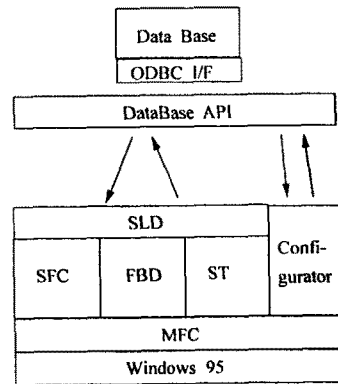


그림 2. 전체 소프트웨어 구성도

3. 프로그램의 구성

3.1 전체구성

Soft Logic Designer(이하 SLD)는 크게 4가지 영역, Sequential Function Chart(이하 SFC) 편집기, Function Block Chart(이하 FBD) 편집기, Structured Text(이하 ST) 편집기와 변수와 시스템의 초기 환경을 설정하는 Configurator로 나누어진다. 3개의 편집기에서는 각각의 언어에 작성, 수정하는 일을 담당하고, configurator는 보드의 종류(direct IO, remote IO)

와 보드의 시작주소등 하드웨어적인 초기환경의 설정과 IO 및 내부 메모리 변수를 정의하여 프로그램에서 이용하도록 한다. SFC와 FBD 편집기는 그래픽 언어를 편집하는 툴이므로 ST 편집기와는 다른 형태를 취한다. SFC와 FBD언어 자체도 매우 상이한 특징을 갖고 있으므로 세가지 편집기가 모두 독자적인 편집기를 가지고 있어야 한다.

3.2 Document-View 구조

SLD의 모든 편집기는 MFC 프로그램의 일반적인 형태인 Document-View의 구조(이하 D-V)를 갖는다. D-V구조는 사용자 인터페이스등 화면에대한 처리는 View 클래스에서 담당하고, 내부데이터 처리는 Document 클래스에서 담당하게 하여 객체지향적인 특성을 잘 이용한 구조라 할수 있다. 다음은 D-V 구조에대한 도식적인 표현이다.

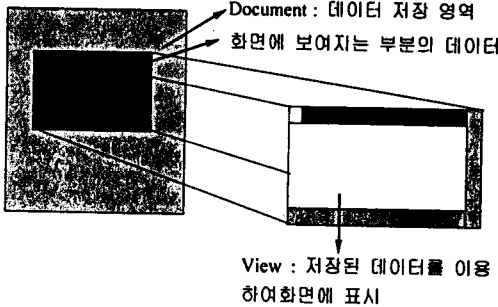


그림 3. MFC Document-View 구조

SLD에서는 SFC, FBD, ST 세개의 편집기가 공히 D-V 구조를 이용하였다. 다음은 SLD 편집기의 클래스 계층도 이다.

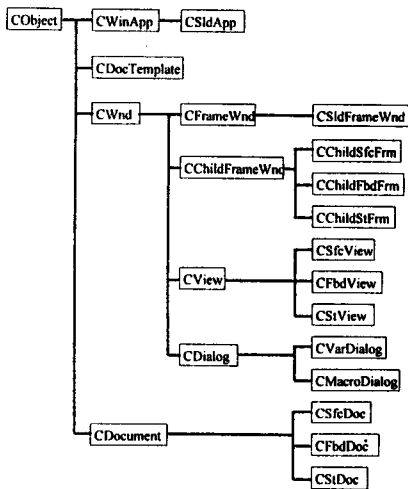


그림 4. SLD 클래스 계층도

3.3 편집기의 기능

제어 프로그램 편집기는 오브젝트의 삽입, 삭제, 이동, 수정, 복사, 되돌리기등 기본 편집기능외에 FBD편집기에서는 다른 편집기에는 없는 외부에서 정의된 변수를 쉽게 입력 할수있는 기능과 매크로 함수를 입력할수 있는 매크로 함수 입력기가 필요하다.

외부변수 입력 기능과 매크로 함수 입력기능은 사용자가 마우스로 메뉴를 선택하고 변수 혹은 프로그램의 본체 부분만을 입력하게 하여 프로그램 작성이 용이하게 하고 프로그램상의 오류를 미연에 방지하여줄 수 있다.

3.4 SFC 편집기

SFC 편집기는 SFC의 특징을 고려하여 구현되었다. SFC는 일반적인 그래픽 편집기와는 달리 고정된 위치에 오브젝트를 놓는다. 화면을 n X n의 배열로 구분하여 해당영역에만 SFC의 오브젝트(STEP, TRANSITION)을 넣을수 있다. SFC의 오브젝트는 직접 데이터(프로그램)를 포함하고 있지 않고 데이터가 작성되어있는 파일의 이름을 갖고 있다. SFC의 오브젝트를 기술하는 데이터 구조는 그림 5와 같은 형태를 지니고 있다.

데이터	데이터형
화면상의 위치	CPoint
오브젝트 이름	CString
오브젝트 번호	int
오브젝트 타입	int
파일 오브젝트	CFile

파일명	CString
파일속성	int
파일내용	CString

그림 5. SFC 데이터 구조

3.5 FBD 편집기

FBD 편집기는 SLD에서 구현된 편집기중 가장 복잡하고 다양한 기능의 편집기이다. FBD 편집기에서는 사용자가 SFC와 같은 정해진 위치에 오브젝트를 삽입하지 않고 화면상의 임의의 위치에 놓을수 있으므로 내부 데이터 구조는 이중 연결 리스트로 구현한다. 이중 연결 리스트는 임의의 위치의 삽입과 삭제가 가능하므로 FBD 편집기와 같은 구조의 데이터를 저장하기에 적합하다. FBD 편집기는 데이터베이스에서 사용자가 정의한 입출력 변수에 대한 정보를 가져와 사용자가 프로그램 작성시 이용할수 있게하는 인터페이스를 제공한다. 또한 사용자가 이미 정의된 블럭이외의 블럭을 사용하고자 할때는 매크로 편집기를 이용하여 편집할수 있다.

FBD 편집기 내의 한개 블럭은 그림 6과 같은 구조를 갖고 있다.

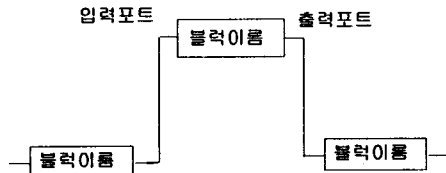


그림 6. 기본 블럭 연결도

그림 6와 같이 FBD 한 블럭이 필요한 정보는 자신에 대한 정보 이외에 입력과 출력포트에 연결된 블럭의 정보를 알아야 한다. 또한, 특별한 구조를 갖는 블럭으로 입력, 출력 변수 블럭이 존재한다. 이는 입출력 변수에 대한 정보를 담고 있어 다른 블럭과 구분되나 전체 FBD 구조에 포함시켜 구현되었다. FBD 데이터 구조의 내부는 그림 7에 나타내었다.

데이터	데이터 형
블록 영역	CRgn
블록 이름	CString
블록 번호	int
블록 입력 포트	CInNode (그림 8)
블록 출력 포트	COutNode (그림 9)
총 입력 갯수	int
총 출력 갯수	int
블록 타입	int

그림 7. FBD 데이터 구조

데이터	데이터 형
index	int
연결된 블록 포인터	CFbdData *
입력선 이름	CString

그림 8. 입력 포트 데이터 구조 (CInNode)

데이터	데이터 형
index	int
연결된 블록 포인터	CFbdData *
연결선 정보	CPoint
총연결선 갯수	int

그림 9. 출력 포트 데이터 구조 (COutNode)

3.6 ST 편집기

일반적인 텍스트 편집기로 편집기 화면은 에디트 공용 컨트롤을 사용하여 구현되었다. FBD 편집기로 작성하기에는 많은 분량의 데이터나 이미 ST로 작성되어있는 프로그램의 일부는 편집기의 입력, 수정, 삭제, 복사 기능을 이용하여 작성할 수 있다.

3.7 구현 화면 예

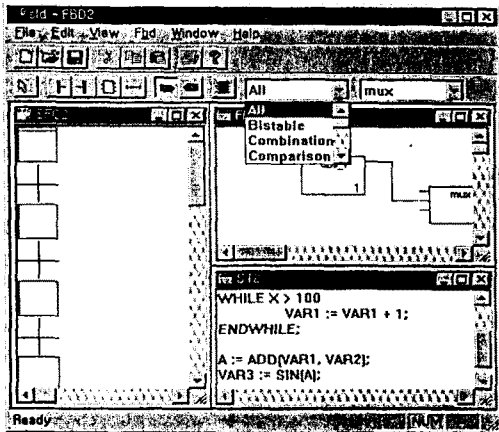


그림 10. SLD 전체 화면

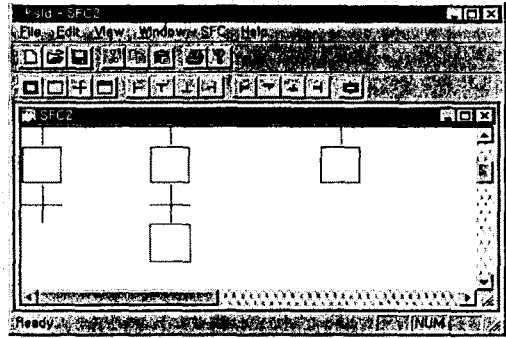


그림 11. SFC 편집기 화면

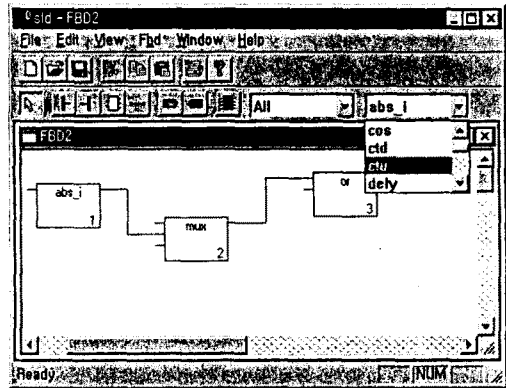


그림 12. FBD 편집기 화면

4. 결론

본 논문에서는 IEC1131-3규격에 입각한 SFC, FBD, ST 3가지의 프로그래밍 언어를 이용하는 제어 시스템 설계 프로그램에 대하여 전체적인 구조와 구현예를 보였다. 본 논문에서 보여준 프로그램 설계물은 원도 95환경하의 일반적인 응용 프로그램과 같은 사용자 인터페이스를 제공하여 원도95를 사용해본 경험이 있는 사용자라면 쉽게 접근할 수 있으며, 직관적인 GUI 환경을 적용하여 마우스와 키패드의 버튼을 이용하여 누구나 쉽게 프로그램을 작성할 수 있게 하였다. 향후 5개의 모든 프로그램 언어에 대한 구현 및 온라인상에서 프로그램의 실행상태를 모니터링할 수 있는 디버거가 추가되면 보다 활용범위가 넓어질 수 있다.

5. 참고문헌

- [1] INTERNATIONAL STANDARD IEC1131-3, International Electrotechnical Commission
- [2] Mark Maslar, *IEC1131-3 Standardizes PLC Programming Languages*, Control Engineering, March 1995, pp. 41-45
- [3] Jeremy Pollard, *The Future of PLC Programming*, Control Engineering, February 1995, pp. 83-88
- [4] *Microsoft Visual C++ MFC Library Reference*, Microsoft Press
- [5] *Microsoft Visual C++ User's Guide*, Microsoft Press