

흐름 제어 언어 분석 도구 설계 및 구현

김선주⁰, 김태완, 장천현
건국대학교 컴퓨터 공학과
(begin⁰, twkim, chchang)@konkuk.ac.kr

Design and Implementation of Analysis Tool for Flow Control Language

Sun-ju Kim⁰, Tae-Wan Kim, Chun-Hyon Chang
Dept. of Computer Engineering, Kon-Kuk University

요 약

산업분야에서 자동화 시스템은 자동설계, 생산설비의 관리, 품질검사 등 각종 생산과 관련되어 모든 일을 자동으로 처리 할 수 있도록 하여 생산성을 향상 시킨다. 일반적으로 자동화 시스템에서 사용되는 소프트웨어는 사용할 수 있는 흐름 제어 언어가 2종류 이하로 제한이 되어 있고, 동일한 시스템에서 언어의 혼용을 통한 시뮬레이션이 불가능하다. 이에 본 논문에서 혼용 사용이 가능한 흐름 제어 언어 통합 분석기를 제시한다. 고급언어 형태의 ST를 기초로 확장한 언어인 EST를 제시하고 통합 분석기를 위하여 그래픽언어를 EST로 변환하고 흐름제어 규칙 작성이 가능한 FBD 편집기, FBD 편집기에서 출력한 EST를 다시 변환하는 EST-IL 변환기를 설계 및 구현한다. EST를 기준으로 그래픽 언어를 통합하는 통합 분석기는 흐름 제어 언어의 통합 시뮬레이션이 가능하다.

1. 서 론

자동화 시스템은 제품의 자동설계, 생산 공정의 자동 제어, 생산설비의 관리, 장애의 발견과 복구, 품질검사 등 각종 생산과 관련되어 모든 일을 자동으로 처리 할 수 있도록 하여 생산성을 향상 시켜 준다.

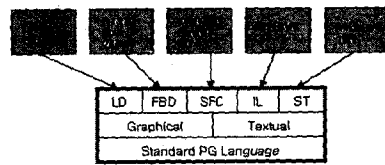
이런 자동화 시스템에 사용 하는 언어는 산업용 표준인 IEC(International Electrotechnical Commission) 61131-3 에서 제안한 IL(Instruction List), ST(Structured Text), FBD(Function Block Diagram), SFC(Sequential Function Chart), LD(Ladder Diagram)가 있다. 그런데 산업 현장에서 사용되는 소프트웨어는 사용할 수 있는 흐름 제어 언어가 2종류로 이하로 제한되고, 동일한 시스템에서 언어 혼용을 통한 시뮬레이션이 불가능하다. 이에 본 논문에서는 흐름 제어 언어에 대한 분석과 EST(Extended Structured Text)를 기반으로 통합원리 구조와 흐름 제어 언어 통합 분석기를 설계 및 구현 한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구로 흐름 제어 언어(IL, ST, FBD, SFC, LD)와 EST 에 대해 설명을 한다. 3장에서는 흐름 제어 언어 분석도구 설계 및 구현에 대해 설명 한다.

2. 관련 연구

2.1 흐름 제어 언어 표준 규약

흐름 제어 언어는 표준 규약인 IEC61131-3 에서 제한한 언어로써 크게 공통 요소와 프로그래밍 언어로 나눈다. 공통 요소에는 흐름 제어 언어에서 범용적으로 사용할 수 있는 데이터 타입과 변수에 대하여 정의되어 있다. 데이터 타입은 논리형, 정수형, 실수형, 문자열형 사용자가 정의한 데이터 타입 등 일반 데이터를 표현하기 위해 다양한 형태로 정의하고 있다 [1]. 프로그래밍 언어는 (그림 1)에서와 같이 그래픽형태의 언어인 LD와 FBD, 텍스트형태의 언어인 IL과 ST, 그리고 그래픽형태와 텍스트형태의 요소를 모두 가지고 있는 SFC로 구성되어있다[2].

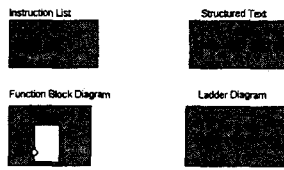


(그림 1) IEC 61131-3 표준 언어

2.2 흐름제어 언어

제어 시스템에 사용되는 프로그래밍 언어로 IL, ST, FBD, SFC, LD 가 있다. IL은 컴퓨터 언어의 어셈블리 언어와 유사한 형태를 취하며, 연산자, 변경자, 피연사자 등의 단순화된 언어 형태를 취하고 있다[4]. ST는 컴퓨터 언어인 Pascal 또는 Basic과 매우 유사한 형태의 고급 언어이다. 컴퓨터 언어에서 기본 구조인 조건, 순환 처리에 대한 내용을 모두 포함하고 있다[4]. FBD는 산업 처리용에서 흔히 사용되는 것

으로 기능과 기능의 동작으로 표현되며, 제어 요소들 간의 정보나 데이터의 흐름을 나타내기에는 적합하다. 주기적인 작업 수행을 정의하기 위해 사용되는 언어이다. 또한 전자 회로처럼 그래픽 블록형태의 조합을 취하고 있다[4]. LD는 코일이나 접점 등의 릴레이 로직을 그래픽 형태로 표현하며, 모션, 연결선, 접점, 코일 등의 구성요소를 보유한다[4]. SFC는 타 언어에서 표현된 작업들에 대한 상위 구조 개념을 가지며, 전체 수행에 대한 흐름 정의에 적합하다. 구성요소로는 스텝, 천이, 액션, 액션 제한자, 흐름선 등이 있다[4].

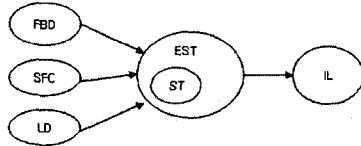


(그림 2) IL, ST, FBD, LD

2.3 EST(Extended Structured Text)

EST언어는 ST, FBD, LD, SFC를 IL로 변환하기 위한 중간 언어이다[5]. 3종 언어(FBD, LD, SFC)를 ST로 표현할 수 없는 부분을 ST를 확장한 EST로 새롭게 정의한다.

EST는 ST기반에서 그래픽 형태의 언어 FBD, LD, SFC 정보를 표현하기 위하여 확장된 언어이다[5].

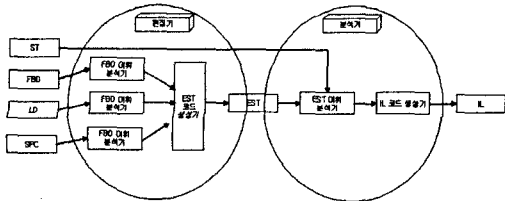


(그림 3) 4종 언어- EST 확장

3. 흐름제어 언어 분석 도구 설계 및 구현

3.1 전체 구조 및 특징

흐름제어 언어 분석의 전체 구조는 FBD 편집기 부분과 EST-IL 변환기 부분으로 나눌 수 있다. FBD 편집기 부분은 FBD 데이터를 그림으로 출력하는 부분이다. FBD 편집기상에서 FBD, LD, SFC는 EST코드 생성기를 거쳐 EST로 변환된다. 변환된 EST는 EST-IL 변환기를 거쳐 최종적으로 IL로 변환된다. EST 코드를 생성하기 위해서는 FBD, LD, SFC와 EST간의 매핑 테이블을 이용한다. 또 IL로 변환하기 위해서는 EST와 IL간의 매핑 테이블을 이용한다.



(그림 4) 전체 구조

3.2 FBD-EST 매핑 테이블

FBD와 EST 변환 하는데 공통으로 사용되는 연산자, 함수, 함수 블록간의 변환 규칙을 다음 표와 같이 나타낸다. (표 1, 2)은 함수와 함수 블록에 대한 변환규칙을 나타낸 것이다. 입력 변수와 출력변수의 갯수에 따라 분류가 된다[5].

(표 1) FBD - EST 함수 변환 관계

구분	FBD	EST
1		B:=COS(A);
2		C:=MAX(A,B);
3		D:=INSERT(A,B,C);
4		F:=MUX4(A,B,C,D,E);

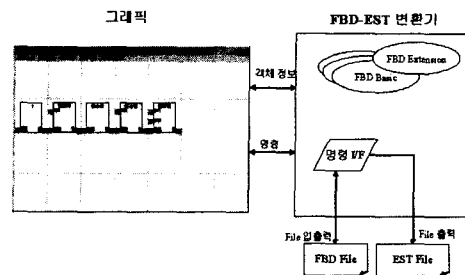
(표 2) FBD-EST 함수 블록 변환 규칙

구분	FBD	EST
1		R_TRIG(A); B:=R_TRIG.Q;
2		RS(A,B); C:=RS.Q1;
3		CMP(A,B); LT:=CMP.LT; EQ:=CMP.EQ; GT:=CMP.GT;
4		CTD(A,B,C); D:=CTD.Q; E:=CTD.CV;

3.3 FBD 편집기

FBD 편집기는 사용자가 흐름제어 규칙에 따라 그림을 작성후, EST코드로 처리하는 기능을 가진다.

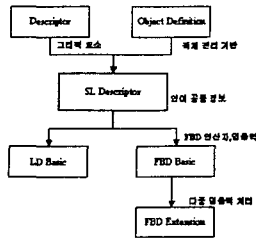
FBD 편집기는 FBD 그래픽과 FBD-EST 변환기로 나눌 수 있는데, FBD 그래픽 부분에서는 그래픽에 좌표, FBD 속성, 입출력, 파일 입출력, 이벤트 등을 정의한다. FBD-EST 변환기 부분은 그래픽정보를 FBD-EST 매핑 테이블에 정의된 내용에 따라 EST코드로 변환한다. 또, FBD 그래픽 명령에 의해 FBD-EST의 인터페이스를 호출함으로써 FBD 그래픽과 데이터 구조를 분리한다.



(그림 5) FBD 편집기

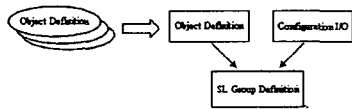
FBD 편집기에서 표현되는 하나의 FBD 객체는 단위 객체 구조로 표현이 된다. (그림 6)에서 Descriptor는 그래픽 언어(LD, FBD, SFC)가지고 있는 공통적인 특징을 가지고 있다. SL Descriptor에서 그래픽언어의 공통 정보를 가지고 있다.

이에 FBD 언어 개발 후 LD, SFC 구조는 단위 객체 구조의 상속만으로 처리 가능하다.



(그림 6) 단위 객체 구조

FBD-EST변환기에서는 단위 객체는 삽입, 삭제, 추가하기 위한 그룹 관리가 필요하다. 그룹으로 관리 위해서 Object가 연결이 되는 것이기 때문에 Object Definition이 정의하고 이를 기초로 그룹관리 기능을 수행하는 SL Group definition이 정의되어야 한다.



(그림 7) 그룹 객체구조

3.4 EST-IL 매핑 테이블

FBD 편집기에서 생성된 EST코드를 IL로 변경하기 위해서는 언어간 매핑테이블이 필요하다. EST언어를 IL로 변환 시 연산자와 피 연산자 정보 이외에는 고정된 형태를 취하므로 두 언어간의 테이블로 처리한다. (표 3,4)는 EST-IL 간의 변환규칙을 보여준다.

(표 3) EST-IL-연산자 변환 규칙

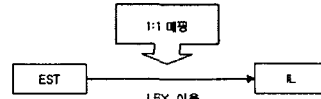
ST	IL	ST	IL
a1 := a2	LD a2 .ST a1	ANY_TO_BOOL(a1)	LD a1 ANY_TO_BOOL
a1 + a2	LD a1 .ADD a2	ANY_TO_DINT(a1)	LD a1 ANY_TO_DINT
a1 - a2	LD a1 .SUB a2	ANY_TO_REAL(a1)	LD a1 ANY_TO_REAL
a1 / a2	LD a1 .DIV a2	ANY_TO_SINT(a1)	LD a1 ANY_TO_SINT
a1 * a2	LD a1 .MUL a2	ANY_TO_TIME(a1)	LD a1 ANY_TO_TIME

(표 4) EST-IL 조건 문 변환 규칙

종류	EST	IL
IF	IF BLOCK1 THEN LOCK2 ENDIF BLOCK3 ENDIF	BLOCK1 JMPN LABEL1 BLOCK2 JMP ENDLABEL LABEL1 : BLOCK3 ENDLABEL:
CASE	CASE Pa OF NPa1: BLOCK1 NPa2: BLOCK2 ELSE BLOCK3 ENDIF	LD Pa EQ NPa1 JMPC LABEL1LEVEL1 LD Pa2 JMPC LABEL2LEVEL1 JMP JAVEL3LEVEL1 LABEL1LEVEL1: BLOCK1 JMP LASTLABEL1LEVEL1 LABEL2LEVEL1: BLOCK2 JMP LASTLAVEL1LEVEL1 LABEL3LEVEL1: BLOCK3 LASTLAVEL1LEVEL1:

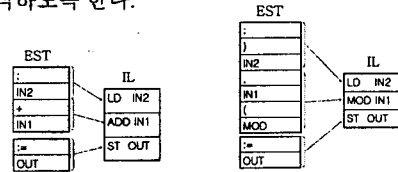
3.5 EST-IL 변환기

EST-IL 변환기는 EST의 어휘를 IL로 어휘로 변환 한다. EST 어휘 분석은 LEX를 사용한다. LEX사용시 연산자와 피 연산자 이동 명령 등을 변환 시 해결하기 위하여 ‘;’ 단위의 문장을 기준으로 스택(Stack)구조에 저장 후 출력 시에 IL코드를 완성한다.

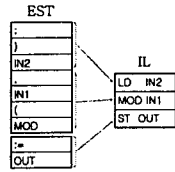


(그림 8) EST-TO - IL 변환

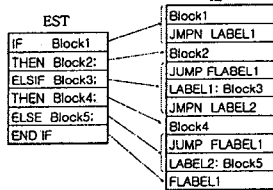
(그림 9,10,11)은 EST 가 IL로 변환되는 것을 보여준다.스택에 push 했다가 다시 pop을 한다. ‘:=’ 이으면 pop을 하고 ST를 스택에 넣는다. ‘;’ 문장의 종료를 뜻하고 pop을 하고 LD를 스택에 넣는다. ‘;’ 뒤에 ID를 체크해서 함수이 아닌 지를 인식하도록 한다.



(그림9)연산자 변환



(그림 10)함수 변환



(그림 11) 조건 문 변환

4. 결론 및 향후 계획

본 논문에서는 흐름제어 언어에 대한 특성과 변환기상에서 작성된 FBD를 텍스트 형태의 언어인 EST로 변경하는 방법과 EST를 IL로 변환하는 방법을 제시하였다. 이를 기초로 FBD-EST 방법을 이용한 FBD 에디터와 EST에서 IL코드로 변환하는 EST-IL변환기를 설계 및 구현하였다. EST를 기준으로 FBD, LD, SFC 언어를 통합하는 통합 분석기 개발은 흐름제어 언어의 통합 시뮬레이션이 가능하다. 향후 계획으로는 실시간 디버깅 기능 개발과 EST을 이용한 시뮬레이션 시스템 개발을 목표로 한다.

참고문헌

- [1]International Standard IEC61131-3, First Edition, 1993
- [2]PLCopen 홈페이지, <http://www.plcopen.org>
- [3]ISaGRAF PRO User Guide 1999
- [4]리얼게인 홈페이지, <http://222.realgain.co.kr>
- [5]정은영, 김선주, 김태완, 장천현, 김문희, 흐름 제어 언어의 통합분석을 위한 확장 ST, 한국정보처리학회 학술발표논문집 9권 제1호 2002, pp. 1013-1016