
SFC언어를 이용한 Elevator 운전 제어회로 설계에 관한 연구

이상문*·김민찬*·곽군평*

A Study on the Design of an Elevator Driving Control Circuit Using SFC Language

Sang-mun Lee* · Min-Chan Kim* · Gun-Pyong Kwak*

이 논문은 2005년도 창원대학교와 과학 기술부·한국과학재단 지정 창원대학교 공작기계기술연구센터의 연구비 지원에 의한 것입니다

요 약

PLC는 자동화 시스템의 발달에 따라 적용 범위가 확대되어왔으며 현재 PLC의 기본 언어 중 LD 언어가 가장 많이 사용되어 지고 있으나 제어 흐름을 나타내는 순차 제어 논리의 기술이나 조건이 많아질 경우 프로그램의 이해가 어려운 단점이 있다. SFC 그래픽 언어는 플로우 차트 형식을 기반으로 하여 순차 제어 논리의 기술에 적합한 언어이며 제어의 흐름을 이해하기 쉽고 유지보수가 용이하여 신뢰성이 향상된다. 본 논문에서는 SFC 언어를 이용하여 3층의 간이 엘리베이터를 동작시키는 제어회로를 설계하고 LD 언어와 비교하여 보았다. 엘리베이터의 운전 제어회로를 설계함에 있어 액션 제한자와 선택 분기문을 사용함으로써 LD 언어를 사용할 때 보다 프로그램 메모리의 용량을 줄이고 처리시간을 단축시켰다.

ABSTRACT

Ladder Diagram(LD) is the most extensively used among PLC standard language for the design of control system. But LD has the disadvantages for data processing and maintenance. On the other hand, the Sequential Function Chart(SFC) graphic language is very powerful for describing the sequential logic control algorithm. SFC is based on flow chart, so control flow understanding is very easy and divergence can possible improving its ability. In this paper, we propose the efficient management elevator system using the action qualifiers and choice divergence. From the result, we confirm the SFC language reduced program memory capacity and processing time is faster than LD language.

키워드

PLC, SFC language, Elevator control, Divergence

I. 서 론

자동화 시스템을 실현할 수 있는 대표적인 기기로

PLC(Programmable Logic Controller) 시스템을 들 수 있다. 1969년 미국의 GM사의 요구에 따라 출현한 PLC는 마이크로프로세서의 발달로 초기의 기본적인 기능

에서 벗어나 아날로그 제어, PID제어, 로봇제어를 비롯한 컴퓨터와의 통신을 통해 산업용 제어의 전 분야에 걸쳐 중요한 역할을 하고 있다. 이런 PLC는 전자요소 산업의 발달 및 컴퓨터 기술의 발전, 생산품의 다품종 소량 생산화 그리고 생산 시스템의 자동화 물질 등의 여러 산업 요소가 서로 맞아 떨어지면서 비약적으로 발전해 왔다. 특히 제어 대상인 기계 장치 프로세스 공정 순서를 미리 정해 놓고 이에 따라 순차적으로 제어하는 순서 제어와 어떤 조건의 상태에 따라서 출력을 내는 조건 제어를 포함하는 의미의 시퀀스 제어를 수행하는 PLC는 단순히 릴레이 시퀀스의 대용품으로 뿐만 아니라 공장 자동화에서 요구되는 분산 및 집중 제어기능, 속도제어, 아날로그 공정제어 등의 고기능화, 소형 및 저가격화로 최소의 제어 시스템에서도 경제적인 적용성 증가 등의 추세로 최근 하드웨어 측면에서 발전 및 연구되어 왔다[1],[2].

PLC는 공장 자동화 (FA :Factory Automation)의 요구에 맞춰 기능이 향상되었고 적용 범위가 확대 되었다. 산업 제어 시스템에서 소프트웨어의 질적인 향상 뿐만 아니라 개발 효율을 높이고 프로그램 기술을 향상시키고자 PLC프로그램 언어의 표준이 나오게 되었다[3]. 표준 언어로는 텍스트 기반의 언어로서 IL (Instruction List), ST (Structured Text)가 있고, 그래픽 기반의 언어로는 LD (Ladder Diagram)과 FBD (Function Block Diagram), SFC(Sequential Function Chart)가 있다.

SFC언어는 1977년에 프랑스에서 개발된 GRAFCET(Grafical Description of Step And Transition of Matching State)에 근거한 국제 표준 언어로서, 플로우 차트 형식의 언어이며, 독일인 페트리가 제안한 페트리네트를 기호로 하고 있다. 이산제어 프로그램에서 순차 제어 논리의 기술에 적합한 강력한 그래픽 언어이므로 제어의 흐름을 이해하기 쉬우며, 유지보수가 용이하고, 프로그램의 기술성이 뛰어나고, 기계 장애의 진단성이 우수하다는 장점이 있다.[4],[5] 특히, 시스템의 병렬 처리는 기능 향상의 핵심이 되므로 병렬 처리가 가능할 수 있게 제어기를 설계하기 위해서는 알고리즘 기술 후 이를 직접 PLC의 프로그램으로 적용이 가능한 SFC 그래픽 언어가 최적의 방법이 될 수 있다.[6] 초기 실 계통 적용 모델은 통일 중공업의 3층 고정밀 머시닝 센터로서 이 공작기계의 공구 교환을

SFC를 이용하여 적용하였다.[7]

따라서 본 논문에서는 SFC의 장점인 병렬 분기를 이용하여 조건의 처리가 까다로운 엘리베이터 운전 제어부를 설계하였고, 실제 엘리베이터에 적용은 위험이 존재하기에 PLC는 LG 산전의 Glofa-GM4를 사용하여 3층으로 간이 제작된 소형 엘리베이터에 적용시켜 구동하여 보고 정상 동작 여부를 확인하였으며 LD와 비교하였다.

II. PLC 프로그래밍 언어

PLC의 프로그램을 작성하기 위해서 사용하는 방식에는 크게 니모닉(Mnemonic)과 래더 다이어그램(LD)이 있다. 이밖에도 논리기호 방식, Flow Chart, Time Chart 및 Step Ladder 방식 등이 있으나 이번 절에서는 니모닉 방식과 LD, Time Chart 방식에 대해서 알아보고 다음 장에서 Flow Chart 방식인 SFC 언어의 특징에 대해서 살펴본다.

2.1 니모닉(Mnemonic) 방식

니모닉은 어셈블리 형태의 문자 기반 언어로서 이 방식은 핸디 로더 등을 이용할 때 사용에 편리하나 프로그램 디버깅 등이 어려운 단점을 가지고 있다.

2.2 ST(Structured Text) 방식

ST언어는 Pascal과 유사한 구조를 갖는 고급(High Level) 언어이다. 산업 제어 분야를 위해 특별히 개발된 언어로서, 특히 복잡한 수식 계산이 필요한 시스템에서 유용하다.

2.3 LD(Ladder Diagram) 방식

래더 다이어그램은 도형 기반 언어로서 현재 가장 많이 사용되어 있는 방식으로서 래더 다이어그램을 그리기 위해서는 편집하기에 편리한 컴퓨터가 필요하므로 소규모 PLC에서는 비용이 많이 드는 단점이 있다. 그러나 래더 다이어그램은 회로를 눈으로 직접 확인 등이 가능하여 회로의 변경이나 디버깅 등이 쉬우며 니모닉 방식과 상호 변환이 가능하여 명령어 방식의 프로그램으로도 쉽게 볼 수 있다. 그러나 조건이 많아지게 되는 경우 프로그램의 길이가 너무 길어지

게 되므로 프로그램의 이해가 어렵고 유지보수 또한 힘든 단점이 있다.

2.4 Time Chart 방식

래더 다이어그램과 같이 이용하여 래더 다이어그램 내의 점점이나 출력이 입력에 대해서 어느 시점에 동작하는지를 나타내는데 많이 이용한다. 이 방식을 이용하면 래더 다이어그램의 분석을 쉽게 할 수 있는 이점이 있다.

2.5 FBD(Functional Block Diagram) 방식

FBD언어는 프로그램의 기능적 요소를 블록으로 표현하지만 다른 언어와 직접적인 변환이 이루어지지 못하는 단점이 있다.

2.6 프로그램 언어의 비교

PLC 프로그램을 기술하는 각각의 프로그램들은 표현의 형식이 다양한 만큼 각각의 장, 단점이 있으며, 사용자가 제어하려는 대상 시스템의 요건에 맞추어서 선택하면 된다. 표 1에 대표적인 세 가지 언어를 비교하였다.

표 1. 프로그램 언어의 비교
Table 1. Comparison of Program Language

	LD	Mnemonic	SFC
조건 처리	우수	나쁨	나쁨
순차 제어	보통	나쁨	우수
데이터 처리	나쁨	우수	보통
유지 보수	나쁨	나쁨	우수

III. SFC 언어의 이해

SFC(Sequential Function Chart)는 종래의 PLC 언어를 이용하여 응용 프로그램을 실행 처리 순서에 따라 플로 차트(Flow Chart) 형식으로 전개하는 구조화 표현 방식 언어이다. 기계장치나 프로세스 공정의 순차적인 동작 상태를 그래프 즉 플로 차트로서 표현한 언어이므로 일반 컴퓨터의 소프트웨어 표현 방법과 유사하여 조건이나 데이터 처리의 흐름이 직관적으로 이해되며 제어 기술자가 아니더라도 이해가 용이한 이점이 있다.

3.1 SFC 언어의 개요

SFC는 어떤 전체에서 세부로 Top-Down 시켜 시스템을 전개할 수 있기 때문에 설계, 보수가 용이하고, 2차원의 표현이므로 제어 처리의 흐름이 시각적으로 이해되며, 시퀀스의 진행에 필요한 조건이 명확할 뿐만 아니라 운전 중에도 현재의 제어 상황을 한 눈에 파악할 수 있다.

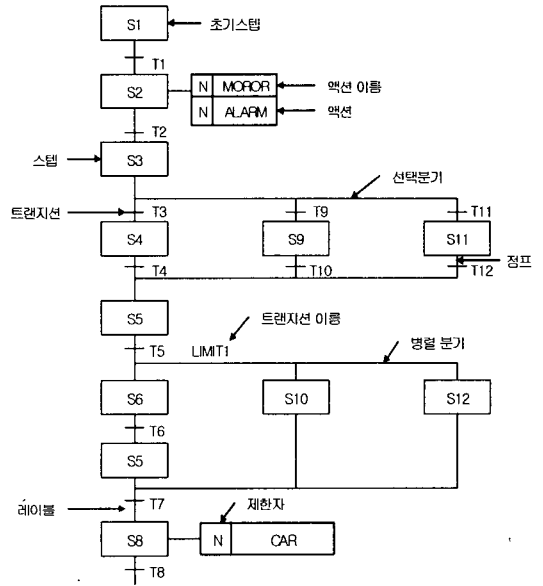


그림 1. SFC 언어의 구조
Fig. 1. Structure of SFC Language

전체의 흐름을 SFC로 기술하고 상세 부분은 IL, ST, LD, FBD 등의 각 언어를 사용하여 기술할 수 있다. 따라서 국부적인 변경은 하나의 Sub-System 중에서 완결하고 다른 시스템 부분에 영향을 미치지 않는다. SFC에서는 래더 다이어그램과 같이 매 Scan 동안에 프로그램 전체를 처리하는 것이 아니고 현재의 제어 대상 부분의 프로그램 밖에 처리하지 않기 때문에 실행 처리 속도를 대폭적으로 향상시킬 수 있다.

SFC는 응용 프로그램을 Step과 Transition으로 분할하여 서로 연결하는 방법을 제공하며 각 Step은 Action으로, 각 Transition은 Transition 조건과 연관된다. SFC는 상태 정보를 가지고 있어야 하기 때문에 프로그램 종류 중 프로그램 블록과 사용자 평선 블록만 SFC로

작성할 수 있고 사용자 평선은 작성이 불가능하다. 형태는 그림 1과 같다.

3.2 SFC언어의 구조

1) 스텝(Step)

스텝은 액션이 연결됨으로써 시퀀스 제어의 단위를 나타낸다. 스텝이 활성화 상태이면 부착되어 있는 액션의 내용이 실행되며 초기 스텝은 최초로 활성화되는 스텝이다. 최초의 활성화 상태인 초기 스텝(S1)의 다음 이행 조건(Transition Condition)이 성립되면 현재 활성화 상태인 스텝은 비활성화 상태로 되고 다음에 연결된 스텝(S2)이 활성화 상태로 되어 액션이 실행된다.

2) 트랜지션(Transition)

트랜지션은 스텝간의 실행 처리 이행 조건을 나타낸다. 이행 조건은 PLC 언어인 IL 또는 LD로 또는 직접 변수로 표현할 수 있고 이행 조건의 결과는 항상 BOOL로 되어야 하며 그 변수의 이름은 어느 트랜지션에서나 TRANS로 사용하여야만 한다. 이행 조건의 결과가 1일 경우 현재 스텝은 비 활성화되고 다음 스텝이 활성화 된다. 스텝과 스텝사이에는 반드시 트랜지션이 있어야 하며 TRANS 변수는 내부적으로 선언된 변수이고 모든 트랜지션에서 이행 조건을 TRANS 변수로 출력시켜야 한다.

3) 액션(Action)

각 스텝에는 액션을 2개까지 연결할 수 있다. 액션이 없는 스텝은 대기 액션으로 여겨지며 다음의 이행 조건이 1이 될 때까지 대기 상태가 됩니다. 액션은 PLC언어인 IL 또는 LD로 구성되고 스텝이 활성화 될 동안 액션의 내용이 실행된다. 액션 제한자가 액션을 제어하는 데 사용되며 액션이 활성화 되었다가 비활성화 상태로 될 때 액션에서 실행된 접점 출력은 0으로 된다. 단, Set 출력, 평선, 평선 블록 출력은 비 활성화되기 전의 상태가 유지된다. 액션이 비 활성화되는 순간 이 액션을 포스트 스캔(Post Scan)한 후 다음 스텝으로 넘어간다.

4) 액션 제한자(Action Qualifier)

액션이 사용될 때마다 액션 제한 자가 사용된다. 스텝에 연관된 액션은 지정된 제한 자에 따라 실행 시점

과 시간이 정의된다.

IV. Elevator 제어 시스템

4.1 제어 시스템의 구성

사람이 탑승하는 Elevator 내부를 카(Car)라고 하며, 승강장(Hall)에서 Elevator를 호출(Call)하는 것을 승강 호출(Hall Call), 카 내부에서 원하는 목적지를 입력하는 것을 카 호출(Car Call)이라고 한다. 각 건물에는 건물 전체에 대한 상황을 파악하기 위한 방이 있는데 이곳을 중앙 관제소라 하고 중앙 관제소 요원에 의해 감시와 제어가 이루어진다.

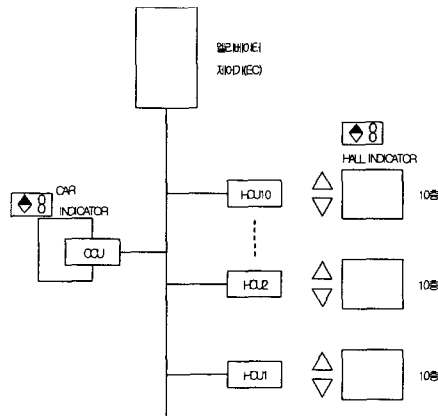


그림 2. 단독 엘리베이터 시스템 구성
Fig. 2 Configuration of Single Elevator System

일반적인 단독 Elevator System의 구성은 그림 2와 같다. 단독 엘리베이터 제어기(EC : Elevator Controller)는 주 제어기(MCP : Main Car Processor)와 모션 제어기(MCU : Motion Control Unit), 위치 탐색기(ADCP : Advanced Car Position Detector), 안전 검사 장치(SLCU : Safety Logic Check Unit), 통신 전용 프로세서(ECP : Elevator Communication Processor)등으로 구성된다.

그리고 Elevator 제어 시스템은 공장 자동화 시스템과 유사하게 제어 구조가 계층적으로 이루어져 있으며 그림 3과 같이 모터 구동부, 속도 제어부, 운전 제어부등으로 나뉘며, 크게 Elevator 한 대의 운전 시퀀스와 모터를 제어하기 위한 Elevator 제어부와 빌딩 내에 인접되게 설치된 여러 Elevator들의 서비스 효율을

극대화하기 위해 이들을 그룹으로 묶고 각 Elevator에서 서비스를 최적으로 분담시키는 군 관리 제어부 등으로 구성된다.

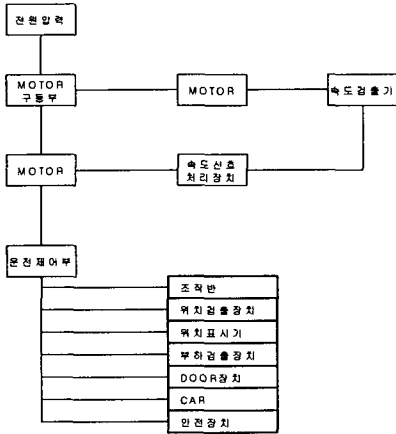


그림 3. 엘리베이터 제어 계통
Fig. 3 Control System of Elevator

4.2 Elevator의 운전 제어부

Elevator 한 대를 운전하는 데 필요한 입출력 신호는 승강가의 속도와 건물의 층수에 따라 다소 차이가 있으나 표 2와 같이 200여개 이상이 된다. Elevator 운전 제어부는 운전에서 필요한 각종 입출력 신호들을 지속적으로 감시하면서 입출력 상태의 조합에 따른 운행 규칙에 의거해, 그에 맞도록 제어하는 일을 담당하게 되는데 다음과 같다.

표 2. Elevator 운전을 위한 입/출력 요약
Table 2. Input/Output Summary for Elevator Drive

구분	종류
안전 진단 및 제어	리미트 스위치, 전원체크, 브레이크, 과속체크, 출입문 개폐 스위치
주행 및 착상 제어	모터 속도 및 카 위치 측정기, 부하 측정기, 속도 명령, 부하 보상
출입문 제어	출입문 개폐 상태, 출입문 개폐 명령, 출입문 개폐 리미트 스위치
고장 감시 및 처리	온습도 센서, 소음 진동 센서, 인버터 고장, 전원 이상 인터럽트
승객 및 승무원과의 인터페이스	호출 버튼, 호출 및 동작 표시, 카의 위치와 방향, 도착 예보 표시
외부 장치와 인터페이스	그룹 제어기, 간이 조작용 단말 장치, 진단 및 수리용 단말 장치

4.3 Elevator 운전 제어부 Flow Chart

운전 제어부의 신호 처리의 흐름을 그림 4에 나타내었다. 먼저 외부의 호출 신호를 처리한 후에 메인 모터의 운전을 지령하고 목표한 층에 도달하면 도어 모터의 운전을 지령한 후 다시 내부 신호의 호출을 처리하는 형식으로 작성되었다. 이 운전 제어부의 흐름을 기본으로 하여 SFC 언어로 PLC프로그램을 작성하였다.

V. 실험 및 시뮬레이션

입력 접점으로는 Glofa-GM의 16접점용을 사용하였고 9번, 18번은 입력의 전원 측과 연결하고 19번 20번은 No-Connect로 둔다. 입력 조건으로는 외부에서 각 층으로 호출시키는 버튼과 카 내부에서 각 층으로 이동 지령을 내리는 내부 버튼, 그리고 각 층의 도착을 감지해내는 리미트 스위치 등으로 구성된다.

입력 접점에 관련된 정보는 표 3과 같다. 출력 접점으로는 Glofa-GM의 16접점용을 사용하였고, 9번, 18번은 출력의 전원 측과 연결하고 19번 20번은 각각 전원으로 연결시킨다. 메인 모터와 도어 모터의 정/역회전 지령과 운전 상태를 표시하는 부분과 현재 카의 위치를 표시하는 부분으로 구성되었고 출력 접점에 관련된 정보는 표 4와 같다.

표 3. 입력 어드레스 정보
Table 3. Information of Input Address

입력 접점	기호	기능
%IX0.0	비상정지	고장시 비상정지
%IX0.0.1	하한리미트	고장시 사고 예방
%IX0.0.2	상한리미트	고장시 사고 예방
%IX0.0.3	일층상향푸쉬	외부 푸시 버튼 : ▲
%IX0.0.4	이층상향푸쉬	외부 푸시 버튼 : ▲
%IX0.0.5	이층하향푸쉬	외부 푸시 버튼 : ▼
%IX0.0.6	삼층하향푸쉬	외부 푸시 버튼 : ▼
%IX0.0.7	도어열림리미트	
%IX0.0.8	도어닫힘리미트	
%IX0.0.9	도어열림푸쉬	내부 도어 열림 버튼 : ◀▶
%IX0.0.10	일층푸쉬	내부 푸시 버튼 : ①
%IX0.0.11	이층푸쉬	내부 푸시 버튼 : ②
%IX0.0.12	삼층푸쉬	내부 푸시 버튼 : ③
%IX0.0.13	일층리미트	일층 도착 감지
%IX0.0.14	이층리미트	이층 도착 감지
%IX0.0.15	삼층리미트	삼층 도착 감지

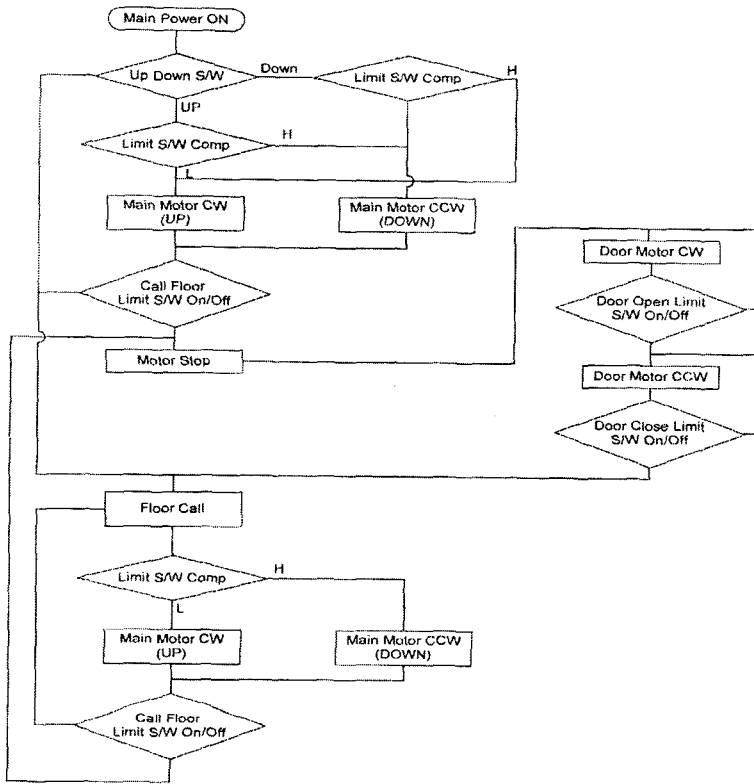


그림 4. 운전 제어부 플로 차트
Fig. 4 Flow Chart of Driving Control Part

표 4. 출력 어드레스 정보
Table. 3. Information of Output Address

출력 접점	기호	기능
%QX0.2.0	하향한계	하한계 경고램프(적)
%QX0.2.1	상향한계	상한계 경고 램(청)
%QX0.2.2	부저	부저 ON
%QX0.2.3	일층표시	현재 위치 표시 : ①
%QX0.2.4	이층표시	현재 위치 표시 : ②
%QX0.2.5	삼층표시	현재 위치 표시 : ③
%QX0.2.6	상향운전표시	운전 상태 표시 : ▲
%QX0.2.7	하향운전표시	운전 상태 표시 : ▼
%QX0.2.8	일층상향표시	외부 버튼 표시 : ▲
%QX0.2.9	이층상향표시	외부 버튼 표시 : ▲
%QX0.2.10	이층하향표시	외부 버튼 표시 : ▼
%QX0.2.11	삼층하향표시	외부 버튼 표시 : ▼
%QX0.2.12	도어모터정회전	도어 열림
%QX0.2.13	도어모터역회전	도어 닫힘
%QX0.2.14	메인모터정회전	카 상향 운전
%QX0.2.15	메인모터역회전	카 하향 운전

5.1 SFC 언어로 설계한 PLC 프로그램

운전 제어부의 Flow Chart를 기본으로 하여 프로그래밍 하였고 층간의 신호처리는 선택 분기를 이용하여 작성하였고 점프를 사용하여 모든 신호에 관련된 처리가 끝나면 다시 처음의 대기상태로 복귀하도록 하였다. 각 트랜지션의 조건이 1이 되게 하는 값은 이전 액션의 동작이 완료되는 조건을 부여하여 스텝에 붙은 액션의 내용이 전부 수행이 되고 나면 다음 스텝으로 자동적으로 넘어 갈 수 있게 하였다.

에러 관리는 Set 액션 제한 자를 사용하여 스텝이 비활성화 되더라도 그 스텝의 액션 내용은 항상 실행하게 하여 언제나 에러를 감시하고 그에 따른 처리를 할 수 있도록 하였다. 또한 인터록에 관한 조건도 Set 액션 제한 자를 이용하여 인터록 사용 횟수를 줄였다. 선택 분기를 사용하여 각 층의 외부 호출에 대한 처리와 내부 호출에 대한 처리를 함으로써 먼저 호출된 층으로 이동을 시키고 이동시키는 동안에도 이동내의

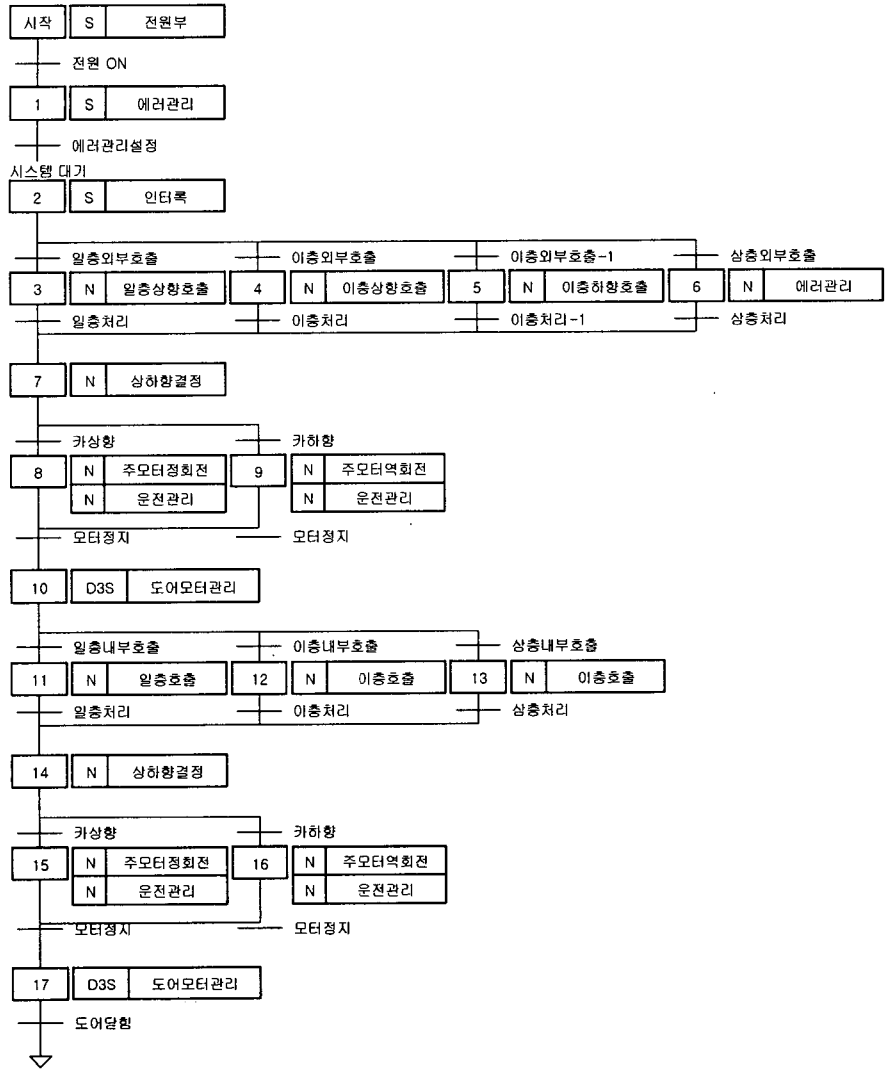


그림 5. SFC로 작성된 PLC 프로그램
Fig. 5 PLC Program with SFC Language

필요 호출 처리나 그 이외의 호출에 관한 처리에도 적절히 대응할 수 있도록 하였으므로 고층의 엘리베이터 시스템에도 적용이 용이하다.

5.2 LD 언어와의 비교

LD언어으로써 3층의 간이 엘리베이터를 제어하기 위해서는 총 84 Line의 프로그램이 필요하게 되고 5Page의 분량이 되었다. SFC 언어로 작성 시 18개의 스텝과 22개의 액션과 각 스텝을 이어주는 트랜지션으로 구

성이 가능하였고 2 Pgae 분량으로 축소될 수 있었다. Page 수가 줄어들음에 따라 전체 프로그램에 대한 이해가 쉬워지고 유지 보수나 기능 추가 등에서 이점이 있다. 프로그램 파일의 크기는 줄어들음을 볼 수 있다. 각각의 프로그램 크기에 대한 비교는 그림 6에 나타내었다. 그림 6의 첫 번째 그림이 LD로 작성한 것이고 두 번째 그림이 SFC로 작성한 것이다.

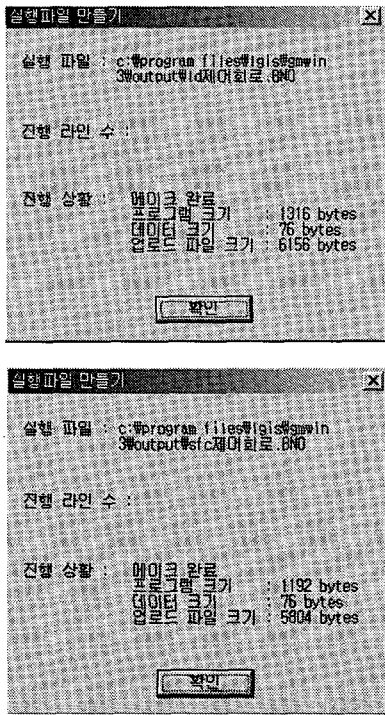


그림 6. LD 언어와의 비교
Fig. 6 Comparison with Ladder Diagram

VI. 결 론

본 논문에서는 산업 현장에서 자동화 공정의 필수 품인 PLC중에서 LG 산전의 Glofa-GM 시리즈를 기본으로 하여 SFC 언어를 이용하여 3층으로 간이 제작된 엘리베이터를 구동시키기 위한 엘리베이터 제어 회로를 설계하였다.

현재까지 PLC의 프로그래밍 언어로는 래더가 가장 많이 사용되어지고 있지만 조건이 많은 경우에 기술이 어렵고 유지 보수가 힘든 단점을 보완해줄 수 있는 SFC 언어로서 엘리베이터 제어 시스템 중 운전 제어부를 기술하였다. 선택 분기를 사용함으로써 각 층에서의 호출 처리나 내부 신호 호출 처리에 있어서 선택 분기의 가지 수만 증가시킴으로써 프로그램의 가독성을 증가시킬 수 있다. 또한 유지 보수가 용이하며 래

더만으로 운전 제어 부를 프로그래밍 하였을 때 보다는 프로그램 크기가 10% 감소하였고, 업로드 파일의 크기도 6% 감소하였다. 여기에 도어의 관리 부분이나 여러 관리 등을 효율적으로 기술할 수 있다면 프로그램의 크기는 더 줄어들 것으로 보인다. 프로그램의 크기가 줄어들고 SFC 언어의 특성상 실행 처리 속도 부분에서도 강점이 있으며 제어의 흐름이 한 눈에 파악할 수 있어 시스템 설계와 보수 시 신뢰성이 향상된다.

본 논문에서는 일반적인 개념의 PLC 프로그래밍 언어로 SFC를 사용해 단독 엘리베이터 시스템을 설계하였지만, 지능형 알고리즘을 도입하여 군 관리 제어 시스템 등으로 확장시킬 수도 있을 것이다.

참고문헌

- [1] I. G. Warnock, "Programmable Controllers Operation and Application", Prentice-Hall Inc, 1998
- [2] Alocca, J. A. and Stuart, "A. Transducer : Theory and Applications", Restion Publishing Inc., Reston, V. A, 1984.
- [3] Masaharu Oku, Takao Kokubu. Shigeru Masuda. and Kenzo Kamiyama, " Application of the Encapsulated Actuator Model to the Sequential Control Machines". Short Papers IEEE/ASME Transaction on Mechatronics. Vol.1, No.4, pp.290-294. 1996.
- [4] Robert E. Smith, "Robotic Vehicles will Perform Tasks Ranging from Production Retrieval to Sub-Assembly Work in Factory of Future." Industrial Engineering, pp.62-72, 1983.
- [5] T.Kouthon, M. A Peraldiabd and J. D. Decotignie, "Distributing PLC Control", International Conf. on IEC, IEEE 21st, Vol. 2, pp.1614-1619, 1995.
- [6] 김종부, 유영욱, 이재춘, 임윤식, "PLC 이론 및 실습", 복두출판사, 2002
- [7] "CNC 콘트롤러 기술 기술개발에 관한 연구 : G7 과제 3차년도 최종 보고서" 통상산업부, 과학기술처, pp. 536-613, 1996.

저자약력



이상문(Sang-Mun Lee)

2003년 창원대학교 전기공학과 졸업.(공학사)
2005년 창원대학교 전기공학과 졸업.(공학석사)

※ 관심분야 : ASIC 설계, PLC 제어 시스템



곽군평(Gun-Pyong Kwak)

1982년 고려대학교 전기공학과 졸업.(공학사)
1985년 고려대학교 전기공학과 졸업.(공학석사)
1990년 고려대학교 전기공학과 졸업.(공학박사)

1990년~1997년 LG산전 연구소 CNC팀 팀장 1998년~
현재 창원대학교 전기공학과 부교수

※ 관심분야 : 제어 알고리즘, Motion Controller



김민찬(Min-Chan Kim)

1996년 창원대학교 전기공학과 졸업.(공학사)
1998년 창원대학교 전기전자제어공학과 졸업.(공학석사)
2003년 창원대학교 전기공학과 졸업.(공학박사)

※ 관심분야 : H_{∞} 제어, 슬라이딩 모드 제어, 적응제어, 시스템 모델링