

# Ladder Diagram 과 Instruction List 와의 상호 변환 알고리즘

안재봉, 유지훈, \* 신영민, 송인창

금성산전 주식회사 연구소

## Mutual Conversion algorithm of Ladder Diagram and Instruction List

Che-Bong An, Ji-Hoon Yoo, Yung-Min Shin, In-Chang Song

GoldStar Industrial Systems Co., Ltd. R&D Center

### Abstract

Programmable Logic Controller(PLC) is an automation device run by user's program. Ladder diagram(LD) and Instruction list(IL) are the most widely used language in this PLC. In this paper, mutual conversion algorithm to convert LD to IL and vice versa is described.

#### 1. 서론

공장자동화에 필수적인 기기인 PLC (Programmable Logic Controller) 는 사용자의 프로그램에 의해 운전이 되기 때문에 릴레이로직에 비해 유연성 및 보수성이 월등히 높다. 이들 PLC 프로그램 언어는 IEC ( International Electro-Technical Commission 규격[1]에 의해 크게 텍스트언어(Textual Language) 와 도형언어(Graphic Language)로 분류된다.






현재까지는 텍스트언어 중에서 IL ( Instruction List)가 프로그램 입력의 용이성 때문에, 도형언어 중에서는 LD(Ladder Diagram)이 릴레이로직과의 유사성 때문에 가장 많이 사용되고 있다.

종래에는 프로그램 입력장치의 화면계약으로 인해 LD로 작성한 프로그램을 수작업으로 따로 바꾸어 PLC에 입력하였으나 현재는 화면표시기기 및 PC (Personal Computer) 의 발달로 LD로 프로그램을 작성하여 자동으로 IL로 바꾸어 PLC에 입력하는 추세이다.

본고에서는 시판되는 PLC중 IL을 운전의 기본으로 하는 PLC(IL Based PLC)에서 사용하는 시퀀스명령어를 중심으로 IL와 LD를 상호변환하는 알고리즘에 대해 기술 하였다.

#### 2. 명령어 및 용어정의

IL의 명령어 종류는 매우 다양하지만 LD 변환과 밀접한 관계가 있는 명령어를 보통 시퀀스 명령어라 하며 각 PLC 회사의 명령어가 이름만 약간씩 다르고 내용은 거의 비슷하다. 본고에서 LD와의 상호변환에 이용한 명령어와 명령어를 기술하는데 사용한 용어에 대한 정의를 표1에 나타내었다.[2]

LD의 구성요소는 접점( , , ) 과 출력( , ) 이며 LD에서 사용한 용어에 대한 정의는 아래와 같다.

- ① 마디 : 직렬접점으로만 연결된 LD의 요소
- ② 복합마디 : (복합)마디와 (복합)마디가 병렬또는 직렬로 연결된 마디

ARG(Arithmetic Register)	논리연산을 행하는 Register
MRG(Memory Register)	마디 정보를 보관하는 Register (다단)
AMR(ARG Memory Register)	ARG 정보를 보관하는 Register (다단)
LOAD(I)	ARG ← 접점의 내용 MRG[i] ← ARG, i ← i+1
OR(I)	ARG ← ARG U 접점의 내용
AND(I)	ARG ← ARG ∩ 접점의 내용
ANB	i ← i-1 ARG ← MRG[i] ∩ ARG
ORB	i ← i-1 ARG ← MRG[i] U ARG
MPS	AMR[n] ← ARG, n ← n+1
MRD	ARG ← AMR[n-1]
MPP	n ← n-1, ARG ← AMR[n]
OUT	출력 Coil ← ARG

〈표1〉 IL의 명령어

Table 1. Instruction of IL

- ③ 병렬마디연결 : (복합)마디와 (복합)마디를 병렬로 연결(그림1)
- ④ 직렬마디연결 : (복합)마디와 (복합)마디를 직렬로 연결(그림1)
- ⑤ 접속선 : 접점과 접점을 접속하는 선
- ⑥ LD의 구성 : LD의 구성 Table은 LADDER[r][c]의 2차원 배열로 나타낸다. LADDER[r][c]는 LD의 왼쪽위를 r=0, c=0으로 하여 오른쪽으로 갈수록 c가 증가하고 아래로 내려 갈수록 r이 증가한다. r은 LD의 열의 숫자이고 c는 행의 숫자이다. 한행은 접속선과 접점을 가지고 있다. LADDER[r][c]의 Data구조는

LADDER[r][c].conj : c행r열 위치에 있는 접속선  
LADDER[r][c].cont : c행r열 위치에 있는 접점이다.

접점은 a접점(+)일 경우 '\/'가, b접점(-)일 경우 '/'가 들어간다.

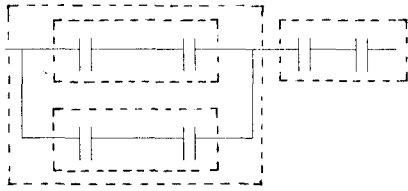


그림1. 직렬 및 병렬연결의 예  
Fig 1. Example of serial and parallel connect

### 3. IL에서 LD로의 변환

주어진 IL에서 LD로의 변환은 IL의 LOAD(I) 명령의 정보를 보관하는 LOAD Table 작성과 실제로 LOAD Table을 보며 LD를 그려나가는 LADDER[][] Table을 작성하는 것으로 나뉜다.

#### (1) LOAD Table 작성

LOAD(I) 라는 명령어는 그림2에 보인것 처럼 LD에서 한개의 마디를 이루며 이 마디들은 ANB 또는 ORB라는 명령어로 각각 직렬 및 병렬로 연결된다. 따라서 IL에서 나타나는 LOAD(I)명령어가 직렬로 연결되는지 병렬로 연결되는지를 조사하여 그 정보를 보관하는 LOAD Table 작성이 필요하다. 이 LOAD Table을 LDT[i]라고 하며 갯수는 프로그램 중에 나오는 LOAD(I) 갯수보다 하나가 적다. (첫 STEP의 LOAD는 Table을 만들 필요가 없다.) LOAD명령어를 검색하여 해당 LOAD가 병렬 접속되면 LDT[i] ← '0', 직렬접속되면 LDT[i] ← 'A' 값을 넣는다.

LOAD Table을 작성하는 Flow를 그림9에, 작성된 예를 표2에 나타내었다.

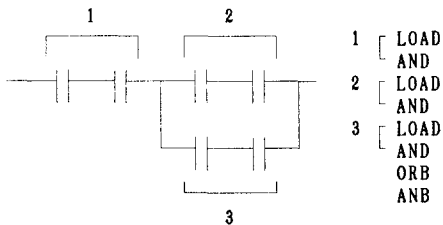


그림2. 마디  
Fig 2. Simple Block

#### (2) LADDER Table의 작성

작성한 LDT[i]와 IL 명령어를 검색하여 실제로 LD를 나타낼수 있는 LADDER Table을 작성한다. LADDER[][]의 정리는 다음과 같다.

사용변수 :

LD\_CNT : LOAD 명령 Counter  
P[index].r : ORB로 연결될 선두위치의 열번호 저장 버퍼  
P[index].c : ORB로 연결될 선두위치의 행번호 저장 버퍼

Instruction	LOAD Table
LOAD X000	
LOADI X001	→ LDT[0] = 'A'
AND Y000	
OR C000	
LOAD M001	→ LDT[1] = '0'
ANDI T000	
ORB	
LOAD B000	→ LDT[2] = '0'
AND L001	
ORB	
ANB	
OUT Y003	

표2. LOAD Table 작성 예  
Table 2. Example of LOAD Table

r : 현재 그리려는 LD의 열 번호  
c : 현재 그리려는 LD의 행 번호

#### A) 첫 STEP에 나오는 LOAD 명령의 처리

```
LADDER[0][0].conj ← 접속선 '/'
LADDER[0][0].cont ← '/' 또는 '\
P[0].r = 0, P[0].c = 0,
index = 1, r = 0, c = 1, LD_CNT = 0
```

#### B) 현 STEP의 명령어가 LOAD(I)인 경우

```
LDT[LD_CNT] = '0' 일때(그림3.㉔)
    i) c ← P[index-1].c
    ii) P[index].c ← c
    iii) r ← 그리려는 열위치 검색
    iv) P[index].r ← r
LDT[LD_CNT] = 'A' 일때(그림3.㉕)
    i) P[index].c ← c
    ii) P[index].r ← r
```

```
i) index ← index + 1
ii) LD_CNT ← LD_CNT + 1
iii) LADDER[r][c].conj ← 접속선
iv) LADDER[r][c].cont ← 접점
v) c ← c + 1
```

#### C) ANB의 경우

```
i) index ← index - 1
```

#### B) ORB의 경우

```
i) index ← index - 1
ii) r에서 P[index-1].r까지 연결선 작도
iii) r ← P[index-1].r
iv) c ← 현 마디의 우측 접속선 위치
```

#### E) AND

```
i) LADDER[r][c].conj ← 접속선(-)
ii) LADDER[r][c].cont ← 접점
iii) c ← c + 1
```

#### F) OR

```
i) c ← P[index-1].c
ii) r ← 그리려는 열위치 검색
iii) r에서 P[index-1].r까지 연결선 작도
iv) LADDER[r][c].conj ← 접속선
v) LADDER[r][c].cont ← 접점
vi) r ← P[index-1].r
vii) c ← 현 마디의 우측 접속선 위치
```

G) MPS

- i) P[index].r ← 그리려는 열 위치
- ii) P[index].c ← 그리려는 행 위치
- iii) index ← index + 1

H) MPP

- i) index ← index - 1
- ii) r ← P[index].r
- iii) c ← P[index].c

I) MRD

- i) r ← P[index].r
- ii) c ← P[index].c

J) OUT

- i) LADDER[r][c].conj ← 접속선
- ii) LADDER[r][c].cont ← OUT coil

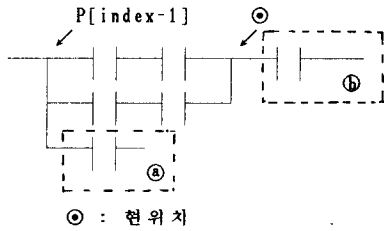


그림3. LOAD의 경우  
Fig3. LOAD case

4. LD에서 IL로의 변환

LD에서 IL로의 변환은 LD 검색위치를 구하고 그 위치의 마디정보 Table을 작성하고 마디를 연결하는 것으로 이루어진다. 그림4에 전체의 흐름도를 나타내었다.

(1) 마디검색 및 마디정보저장

마디를 검색하여 마디에 대한 정보를 B[i]에 저장한다. i번째 마디 정보를 갖는 B[i]의 데이터 구조는

- B[i].row : 마디가 있는 열번호
- B[i].sc : 마디가 시작되는 행번호
- B[i].ec : 마디가 끝나는 행번호
- B[i].so : 마디의 시작행에 병렬접속할 수 있는 마디가 존재하는 열번호.  
병렬접속할 수 있는 마디가 없을 때는 FF Hexa임.
- B[i].eo : 마디의 끝행에 병렬접속할 수 있는 마디가 존재하는 열번호.  
병렬접속할 수 있는 마디가 없을 때는 FF Hexa 임.
- B[i].size : 마디에 포함되어 있는 접점의 갯수

와 같다. B[i]의 구조는 복합마디에 있어서도 적용된다. 그림5에 B[i]를 사용해 마디의 정보를 보판하는 예를 들었다.

마디에 있는 접점의 내용을 IL로 변환시는 마디의 첫 접점을 LOAD(I) 명령어로, 그다음 접점 부터는 AND(I) 명령어로 처리하여 변환시킨다.

OUT을 포함하는 복합마디를 OUT-Block이라고 하며 OUT-Block에서는 B[i]외에 M[i]를 이용해 OUT Block의 정보를 저장한다. 그림6에 대표적인 OUT Block을 나타내었다. OUT Block 정보를 저장하는 M[i]의 구조는

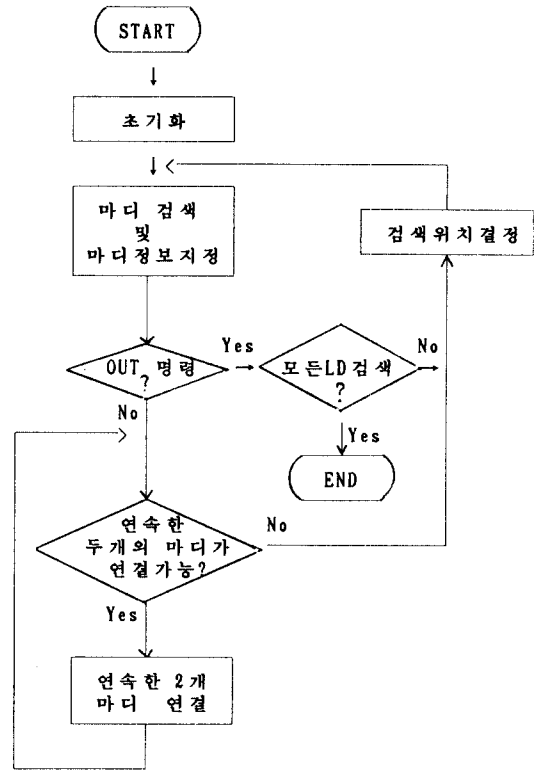
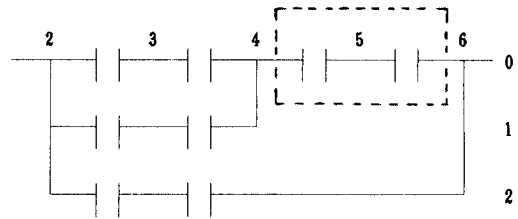


그림4. IL로의 변환 Flow  
Fig 4. Flow of Conversion to IL



- B[i].row = 0
- B[i].sc = 4
- B[i].ec = 6
- B[i].so = FF hexa
- B[i].eo = 2
- B[i].size = 2

그림5. 마디정보의 표현  
Fig5. representation of block information

- M[i].r : OUT을 만났을 때 다음 검색을 시작해야 할 열의 위치이다.
- M[i].c : OUT을 만났을 때 다음 검색을 시작해야 할 행의 위치이다.

OUT BLOCK을 만난 첫 접속선 (①의 위치)은 MPS 명령어가 되며 OUT을 만나고 나서 다음 검색할 위치의 접속선은 MRD 혹은 MPP가 된다. MRD와 MPP의 구별은 다음에 검색할 OUT을 포함한 열이 존재하면 MRD가 되며 (②의 위치) 그러한 열이 존재하지 않으면 (③의 위치) MPP가 된다.

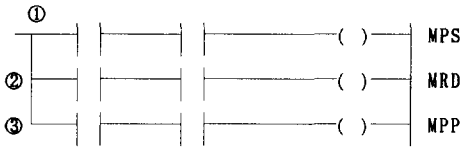


그림6. OUT Block  
Fig.6. OUT Block

## (2) 마디연결 방법

### a) 병렬연결

(복합)마디 B[i] 와 B[i-1]을 병렬연결할 수 있는 조건은

- i) B[i-1].sc = B[i].sc
- ii) B[i-1].ec = B[i].ec
- iii) B[i-1].so = B[i].row
- iv) B[i-1].eo = B[i].row

이다.

상기조건을 만족하는 마디를 병렬연결하기 위해서는

- i) B[i-1].so ← B[i].so
- ii) B[i-1].eo ← B[i].eo

를 행한다.

### b) 직렬연결

(복합)마디 B[i]와 B[i-1]을 직렬연결할 수 있는 조건은

- i) B[i-1].row = B[i].row
- ii) B[i-1].ec = B[i].sc
- iii) B[i].so = FF Hexa

이다.

상기조건을 만족하는 마디를 직렬연결하기 위해서는

- i) B[i-1].ec ← B[i].ec
- ii) B[i-1].eo ← B[i].eo

을 행한다.

## (3) LD의 검색위치 설정방법

LD를 검색하여 IL로 변환하는 과정에서 LD의 검색위치(r,c) 선정은 3가지 조건에 의해 선택된다. 현재 검색되는 연결한 마디의 정보Table을 B[i]라 하면

- i) B[i].eo = FF Hexa인 경우  
r ← B[i].row  
c ← B[i].ec
- ii) B[i].so = B[i].eo인 경우  
r ← B[i].so  
c ← B[i].sc

iii) OUT명령을 만났을때 (현재 검색하는 정보 Table을 M[i]라 하면)

- MPS-MPP Block 이면  
r ← M[i].r  
c ← M[i].c
- MPS-MPP Block이 아니면  
r ← B[i].so  
c ← B[i].sc의 경우이다.

## (4) AND(I), OR(I)로의 변환

AND(I)는 (1)항에서도 변환이 되었지만 현재 직렬 연결한 마디가 복합마디가 아니면 (1)항에서 LOAD(I) 명령어로 취급되었던 것이 AND(I) 명령어로 변환된다.

OR(I)는 현재 병렬연결한 마디의 B[i].size가 1이면 (1)항에서 LOAD(I) 명령어로 취급되었던 것이 OR(I) 명령어로 변환된다. (그림7)

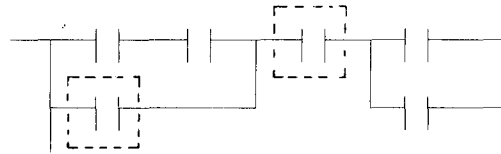


그림7. AND, OR  
Fig.7 AND, OR

## 5. 결론 및 추후 연구과제

본고에서 기술한 변환 알고리즘은 MRG[]를 8단, AMR[]을 8단으로 하여 Personal Computer로 Program 하여 검증하였다. 수행결과 정확한 논리의 모든 LD와 IL을 상호변환 시킬수 있었으며 그림8에 그한예를 보였다. PLC 언어는 LD와 IL을 기초로하여 구조화된 언어를 도입해가는 추세가 발전하고 있다. 추후에는 이런 구조화된 언어와 IL, LD와의 접속에 대한 연구가 추진되어야 하리라 사료된다.

## 6. 참고문헌

- [1] IEC SC 65A/WG6, 65A(Secretariat)67- I, II, III Draft-Programmable Controllers. Part 3-Programming Languages, Feb. 1987.
- [2] 금성계전, 프로그래머블 콘트롤러 HANDY PROGRAMMER LD-40A 취급 설명서, pp. 8-15

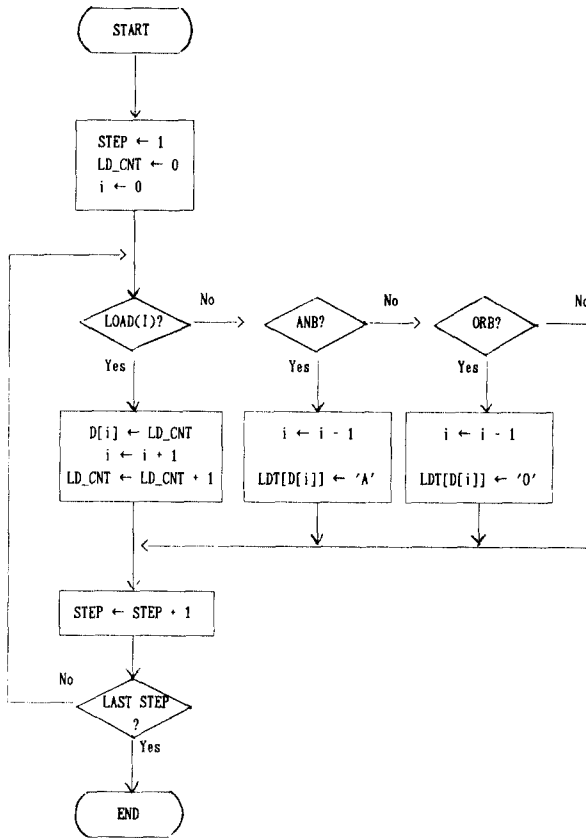


그림 9. LOAD Table 작성 Flow  
Fig 9. Flow of Making LOAD Table

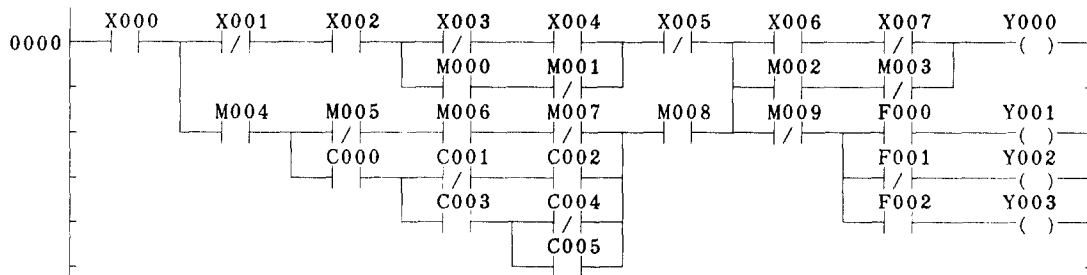
0000	LOAD	X000	0024	ANB	
0001	LOADI	X001	0025	AND	M008
0002	AND	X002	0026	ORB	
0003	LOADI	X003	0027	ANB	
0004	AND	X004	0028	MPS	
0005	LOAD	M000	0029	LOAD	X006
0006	ANDI	M001	0030	ANDI	X007
0007	ORB		0031	LOAD	M002
0008	ANB		0032	ANDI	M003
0009	ANDI	X005	0033	ORB	
0010	LOAD	M004	0034	ANB	
0011	LOADI	M005	0035	OUT	Y000
0012	AND	M006	0036	MPP	
0013	ANDI	M007	0037	ANDI	M009
0014	LOAD	C000	0038	MPS	
0015	LOADI	C001	0039	AND	F000
0016	AND	C002	0040	OUT	Y001
0017	LOAD	C003	0041	MRD	
0018	LOADI	C004	0042	ANDI	F001
0019	OR	C005	0043	OUT	Y002
0020	ANB		0044	MPP	
0021	ORB		0045	AND	F002
0022	ANB		0046	OUT	Y003
0023	ORB				

그림 8.a IL의 예  
Fig 8.a Example of IL

[LADDER DIAGRAM]

DATE : 1990.7.30

PAGE : 001



--- END ---

그림 8.b LD의 예  
Fig 8.b Example of LD