

PLC사다리도편집 및 지령표의 생성을 위한 한가지 방법

리광식, 김용철

위대한 령도자 김정일동지께서는 다음과 같이 교시하시였다.

《프로그램을 개발하는데서 기본은 우리 식의 프로그램을 개발하는것입니다.》

(《김정일선집》 증보판 제21권 42페이지)

우리 식의 PLC를 개발하는데서 PLC통합개발도구를 구축하는것은 중요한 문제로 나선다.

사다리도언어는 PLC프로그램설계에서 가장 일반적인 프로그램작성언어로서 서로 다른 도형부호를 리용하여 여러가지 지령을 나타내며 직렬, 병렬 등의 개념을 가지고 도형부호의 순서위치를 구성하여 조종논리를 표현한다.[2] 그러나 사다리도는 직접 컴퓨터에 의하여 식별되어 실행할수 없으며 그것을 지령표로 변환하여야 실행할수 있다. 사다리도언어를 지령표로 전환하는것은 지령표가 기계어에 근사하기때문이며 이렇게 하면 사용자 프로그램을 CPU가 알수 있는 기계어코드에로의 전환을 더욱 쉽게 할수 있다.

현재 많은 PLC체계개발의 통합환경에서 리용하는 대표적인 호상변환알고리즘으로서 는 AOV도에 의하여 사다리도를 지령표로 변환하는 알고리즘이다.[1] 여기에서는 2진나무의 탐색알고리즘을 리용하여 사다리도와 지령표사이의 호상변환을 완성하는 알고리즘을 제기하였다. 다만 선행연구[1]의 2진나무구조의 변환알고리즘을 리용하면 비록 2진나무로 사다리도의 논리관계를 잘 표현할수 있다고 해도 2진나무의 작성과정은 매우 복잡하다.

본문에서는 이전의 2진나무구조의 변환알고리즘의 부족점을 극복하고 깊이우선원리에 따르는 사다리도를 지령표로 변환하는 최량알고리즘을 제기하였다.

1. 구조론리나무의 구성

우리는 깊이우선순환주사알고리즘을 리용하여 사다리도의 주사방향과 읽기순서를 확정한다.

그 주사방법은 다음과 같다.

① 기준행(매 층의 첫번째 행)의 첫번째 렬로부터 아래방향(Ladder_Wire_Down)선을 포함하는 요소가 발견될 때까지 매 요소를 주사하여 그 요소가 현재 속한 렬(n_Col로 표시)을 기록한다.

② 만일 아래방향선을 포함한 요소가 발견되면 아래행의 현재렬(n_Col)로 넘어가서 그 앞의 렬요소가 빈요소(Element_Type_Free)인가를 판단한다. 만일 빈요소이라면 직접 ③으로, 그렇지 않으면 행련결표(Row_List)를 리용하여 렬관된 행을 보존한 다음 ③으로 이행한다.

③ 현재렬이 아래방향선을 포함하는가를 판단한다. 만일 포함한다면 ②로, 그렇지 않으면 ④로 이행한다.

④ 행렬결표에 기억된 정보에서 현재렬의 앞의 요소가 옷방향선(Ladder_Wire_Up)을 포함하며 그 옷행이 기준행인가를 판단한다. 만일 그렇다면 그 요소가 포함된 행을 분기(Branch)로 하고 분기결표(여기에는 분기시작렬, 끝렬 및 행의 정보 등을 포함하는데 BranchList로 정의한다.)에 기억한다. 그리고 이 과정을 행렬결표의 정보를 전부 읽어낼 때까지 진행하고 ⑥으로 이행한다. 만일 그렇지 않으면 ⑤로 이행한다.

⑤ 앞행의 현재렬이 옷방향선을 포함하는가를 판단하는데 만일 포함하지 않는다면 ④를 실행한다. 만일 포함한다면 계속 앞행이 기준행인가를 판단한다.

⑥ 분기결표에서 읽어낸 기억정보로부터 시작렬과 끝렬의 간격이 1보다 큰가를 판단한다. 만일 크지 않다면 분기결표의 정보를 다 읽어낼 때까지 ⑥을 실행하며 만일 크다면 ①-⑤의 과정을 반복한다.

여기에 기초하여 핵심함수의 주사와 나무를 구조화하는 과정은 다음과 같다.

① 1개의 AND/ANDI마디를 새로 창조하고 왼쪽으로부터 오른쪽으로 주사를 시작한다. 아래방향의 선이 발견되기 전에 주사된 매개 요소들을 AND/ANDI마디의 자식마디로 한다.

② 다음행의 현재렬의 앞렬을 찾아서 그 요소가 빈것인가를 판단한다. 만일 비어있지 않다면 아래방향의 주사를 진행하는데 다음행의 현재렬의 앞렬이 비어있는가를 차례로 판단하여 분기결표 RowList에 련관된 행의 정보를 기억한다. 만일 귀환된 파라메터 RowList가 빈것이라면 주사된 마디점(LDObject)을 AND/ANDI마디점으로 치환한다. 즉 마디점 LDObject는 요소마디점의 아버지마디로 하며 뒤방향주사를 계속 진행한다. 만일 귀환된 파라메터 RowList가 비지 않았다면 ③을 실행한다.

③ 행렬결표(RowList)에서 하나씩 기억정보를 읽어내어 행렬결표안에 기억된 대상이 분기인가를 판단한다. 만일 분기라면 분기정보를 얻어서 분기결표(BranchList)에 기억시킨다.

④ 분기를 주사하여 새로 OR/ORI마디를 창조하고 AND/ANDI마디의 자식마디로 한다. 그리고 OR/ORI마디를 분기함수의 실제파라메터로 하고 분기시작렬 및 끝렬사이에 있는 전체 고리부분에 대하여 주사를 진행한다.

2. 나무탐색으로 대응하는 지령표언기

사다리도의 논리관계를 나무구조로 완성한 후 논리나무마디를 세가지 종류 즉 뿌리마디, 논리마디 및 요소마디로 나누는데 여기서 논리마디는 자식나무와의 결합방식(직렬/병렬)을 나타내며 요소마디는 사다리도안의 구체적인 요소를 나타낸다. 한편 순환탐색알고리즘을 리용하여 뿌리마디의 자식마디에 대하여 차례차례 련결표의 매 대상에 대한 콤파일을 진행한다. 대상이 요소마디일 때 요소마디류형의 콤파일함수를 적재하고 요소의 류형과 논리관계에 따라서 대응한 지령을 생성하며 대상이 논리마디일 때에는 논리마디류형의 탐색함수를 적재하고 그 류형의 함수를 순환하며 논리마디의 자식마디에 대한 련결표를 모두 탐색한 후 논리마디에 대응하는 지령으로 생성한다. 이렇게 부단히 반복하여 련결표안의 모든 대상을 콤파일한다.

콤파일함수의 주요코드는 다음과 같다.

```
bool CLDObject::Compile(OP_CODE_TYPE ocType, int& nAddress, ArrayString*
pstrProgram)
```

```
{
  CLDObjectList::const__iterator it=m__LDObjectList.begin();
  while(it != m LDObjectList.end())
  {
    (*it)->Compile(ocType, nAddress, pstrProgram);
    it++;
    ocType = AND;
  }
}
```

우의 탐색과정을 보면 바로 요소마디와
론리마디의 반복계산과정이라는것을 알수 있다.

체계설계를 완성한 후 그것에 대하여 실험
을 진행하였다.

이를 위해 우선 조종요구에 따라 사다리
도를 작성하고 다음 사다리도에 대응하는 지
령표를 얻었다.

그림은 설계한 PLC사다리도프로그램작성
소프트웨어를 리용하여 작성한 PLC사다리도의 실패이다.

컴파일후에 생성된 지령코드는 다음과 같다.

LD	X001
LD	X002
LD	X003
ORI	X005
ANB	
ANI	X004
LDI	X006
AND	X007
ORB	
ANB	
OUT	Y000

결과로부터 깊이우선순환주사방법이 사다리도프로그램을 효과적으로 주사처리하여
그것을 정확하게 지령표프로그램으로 변환하였다는것을 알수 있다.

맺 는 말

사다리도를 지령표로 변환하는 알고리즘을 개선하여 컴파일속도를 높일수 있는 방법
을 제기하고 실현하였다.

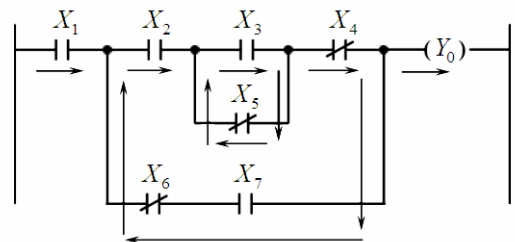


그림. PLC사다리도실패

참 고 문 헌

- [1] 裴锋 等; 广西工学院学报, 21, 3, 42, 2010.
- [2] 毕辉 等; 微计算机信息, 23, 9, 63, 2007.

주체105(2016)년 8월 5일 원고접수

A Method of Editing of PLC Ladder Diagram and Generating of the Instruct List

Ri Kwang Sik, Kim Yong Chol

The design of the editing of PLC ladder diagram and generating of instruct list is proposed, aiming at the limitation and lack of the algorithm of binary tree structure, a new algorithm to convert instruction list to ladder diagram is proposed.

Key words: PLC, ladder diagram, instruct list