

# PLC 래더다이아그램 생성을 위한 지식기반시스템에 관한 연구<sup>†</sup>

## - A Study on the Knowledge-based PLC Ladder Programming System -

강 신 한\*  
김 광 만\*\*  
이 재 원\*\*\*

### Abstract

In this paper, we present the application of knowledge-based system technique for generating of PLC ladder diagram. The developed prototype system receives a time chart as an input and generates a ladder logic as its output. This results in the computerization and intellegent processing of PLC programming. The system can be effectively applied to sequence control where the PLC programs need to be frequently changed and generated.

#### 1. 서론

생산공장의 FA 또는 CIM의 실현은 컴퓨터와 고기능화된 PLC, PLC간의 네트워크, 각종 센서와 고속 정밀도의 드라이브장치 및 이들을 체계적으로 연결하는 통합 소프트웨어로 가능해 지는데, 이 가운데서 센서, 드라이브와 접속하여 실제로 기계를 제어하는 PLC는 매우 중요한 요소중의 하나이다[1]. 이러한 PLC의 프로그래밍작업은 제조회사별로 고유의 프로그램 로더(loader)를 통하여 이루어진다. 그러나 이 방법은 부분적인 편리성은 있지만 몇가지 문제점을 내포하고 있다. 첫째 프로그래머에 의존하기 때문에 발생하는 프로그래밍 파오 및 프로그램의 불균일성이 있고, 둘째 작업부하가 많은 경우 수작업에 기인한 생산성의 저하가 예상되며, 셋째 입출력 제어요소정보의 래더프로그램과의 독립정보관리의 불편함이다. 그리고 넷째로 기존의 프로그래밍 시스템은 이기종 PLC 끼리의 프로그램 호환이 어렵고, 더우기 동일 업체의 경우도 신기종이 발매될 때마다 프로그램로더가 바뀌어 종래의 로더를 사용할 수 없어 기존의 프로그램은 사용할 수 없는 경우도 발생한다. 이상의 단점을 극복하기 위해서는 프로그래머의 전문지식과 PLC의 기능이 함께 이용될 수 있는 방법이 요구된다. PLC 프로그래밍에 관련된 연구동향을 살펴보면 다음과 같다.

Krogh 와 Ekberg 는 사용자가 제어대상 시스템에 대해 고급언어(high level language)형태로 서술하여 주면 서술된 내용을 IEC 규정의 IL(instruction list)형태로 자동생성하여 주는 시스템을 발표하였다[2]. Devanathan은 동작의 순차관계를 제어요소의 ON/OFF 상태로 표현한 STD(state transition diagram)를 이용하여 정보를 입력하는 방법을 사용하여 인공지능 기법을 이용하여 래더로직을 생성하는 시스템을 발표하였다[3]. 이 시스템은 지식의 형태로 규칙(rule)을 이용한 규칙기반시스템(rule-based system)이다. 이 연구로 인공지능기법에 의한 프로그래밍의 자동화 및 유연성이 증대되었다. Gunasena 와 Lehihet는 인터프리터 방식의 정보입력을 시도 하였는데 일정한 포맷을 이용하여 동작간의 전후관계정보, 동작과 관계를 맺는 입출력제어요소, 입출력어드레스 등의 정보가 입력된다[4]. 이 시스템은 인공지능 기법이 아닌 기존의 절차형(procedural) 방식의 변환기(translator)를 이용하여 입력정보를 래더 프로그램으로 변환 생성한다.

† 본 연구의 일부는 인하대학교 '93년도 연구비 지원 및 금성기전(주)의 지원에 의해 수행되었음.

\* 인하대학교 대학원 자동화공학전공 박사과정

\*\* 인덕전문대학 공업경영학과 조교수

\*\*\* 인하대학교 자동화공학과 교수

접수 : 1994년 5월 6일

확정 : 1994년 5월 12일

Mizutani와 그의 연구팀은 동작의 선후관계, 동작과관계를 맺는 구동요소 그리고 구동요소를 제어하는 제어요소를 테이블(table) 형태로 입력하여 내부적으로 의미망(semantic network)을 구축하고 모델을 이용하여 래더다이어그램을 구하는 모델기반추론(model-based reasoning)을 이용하였다[5].

국내에서는 각 제작사별로 하드웨어 및 전용의 래더로더에 관한 연구는 하고 있으나 PLC 프로그래밍 기술에 대한 연구는 매우 빈약하고 특히 인공지능의 지식기반 시스템(knowledge-based system) 기술의 PLC 프로그래밍에의 적용은 거의 없는 상태이다.

본 연구에서는 조립라인을 구성하는 셀의 동작에 관련된 타임차트정보를 입력하면 이를 화일로 저장한 후 PLC 래더다이어그램을 자동으로 생성하는 지식기반 시스템의 프로토타입(prototype)을 개발하였다.

## 2. 지능형 자동화 PLC 프로그래밍 시스템의 기능실제

제품의 생산주기가 짧아지고 제어해야 할 대상이 많아질 수록 조립라인에 대한 정보의 저장 및 재활용은 생산성을 증대시킬 수 있다. 그러나 가장 일반적인 PLC 프로그래밍 방식인 래더로더를 이용하는 경우에는 입력정보가 곧바로 래더다이어그램으로만 표현되어져 구동요소의 동작 순서나 구동요소와 관련된 제어요소에 관한 정보는 별도로 정리되지 않는다. 본 연구에서는 지능형 PLC 자동프로그래밍 시스템이 갖추어야 할 기능으로 다음과 같이 설정하였다.

첫째, 중간래더다이어그램(NLD:neutral ladder diagram) 개념을 도입하여 다양한 PLC 모델에 적용하여 사용할 수 있도록 한다. 이 기능은 래더다이어그램의 링(rung)별 입출력 어드레스 정의가 래더로직 정의와 별도로 정의 되는 것을 뜻하며 생성된 NLD는 그래픽(graphic)으로 출력되고 특정 입출력어드레스가 별도로 정의되도록한다. 이러한 내용은 두개의 정보의 별도관리 뿐만 아니라 하드웨어 기종 변화에 유연한 프로그래밍을 가능하게 하는 것으로 기존의 상용시스템에서 처리하는 방법과 다르다.

둘째, 시스템의 입력정보는 타임차트에 표현되는 정보로 한다. 이 기능은 흔히 산업체에서 사용하는 타임차트의 자연스러운 활용을 고려하기 위한 것이다. 이를 위해 타임차트의 동작전후 순차관계 정보를 텍스트(text) 형태로 시스템에 입력하게 된다.

셋째, 프로그래밍 전문가의 경험이 최대한 반영되는 추론기법을 이용하여 시스템의 효율이 증대되도록 한다. 이를 위하여 본 연구에서는 기존의 경험을 바탕으로 문제를 해결하는 사례기반추론기법을 채택함으로써 이를 반영하였다. 이는 규칙기반시스템(rule-based system)이 갖는 속도문제를 개선해 줄 수 있다[6].

## 3. 조립셀의 구성요소와 타임차트(time chart)

본 연구에서 대상으로 하는 조립셀은 클램프(clamp), 가이드(guide), 착탈기(ejector), 용접건(welding gun), 건래치(gun latch), 브레이크(brake) 그리고 용접로봇(welding robot)등의 구동요소로 구성되며 자동차 조립라인 공정에서 흔히 발견되는 것과 유사하다. 셀의 각 구성요소들은 각종 스위치, 공압실린더 그리고 솔레노이드밸브등의 제어요소에 의해서 순차제어 되어지는데 PLC프로그램은 이 제어요소를 래더다이어그램의 링의 입출력으로 표현하게 된다. 입력요소로는 사람의 개폐동작에 의해 결정되는 조작입력력기기(operating device)와 구성부품간의 상호관계에 의해 인식되는 검출입력기기(sensing device)로 구분되고, 출력기기는 결과를 표시해주는 표시출력기기(displaying device)와 어떤 부품을 실제로 동작시켜주는 구동출력기기(actuating device)로 구분된다.

이러한 조립셀의 PLC프로그래밍작업시 사용되는 자료는 타임차트(time chart)이다. 타임차트에는 작업셀을 구성하는 각 요소들과 이 요소들에 결합되어 있는 입출력기기간의 개폐를 통한 장치들간의 작업 순서와 각 장치에 조립되어 있는 입출력기기가 표시된다[7]. 다음의 Fig.1 은 타임차트를 보여주고 있다. 본 연구에서는 이러한 타임차트의 내용이 시스템의 입력정보가 되도록 하였다.

이와같이 셀의 제어 순서를 표현한 타임차트에 관련된 정보를 대화형으로 시스템에 텍스트형태로 한 줄씩 시스템에 서술하여 입력한다. 이때 입력되는 정보는 셀 구동요소의 종류, 각 구동요소별 동작, 동작간의 순서, 각동작의 개시및 완료 시점, 동작과 관련된 입출력기기에 대한 정보이다.

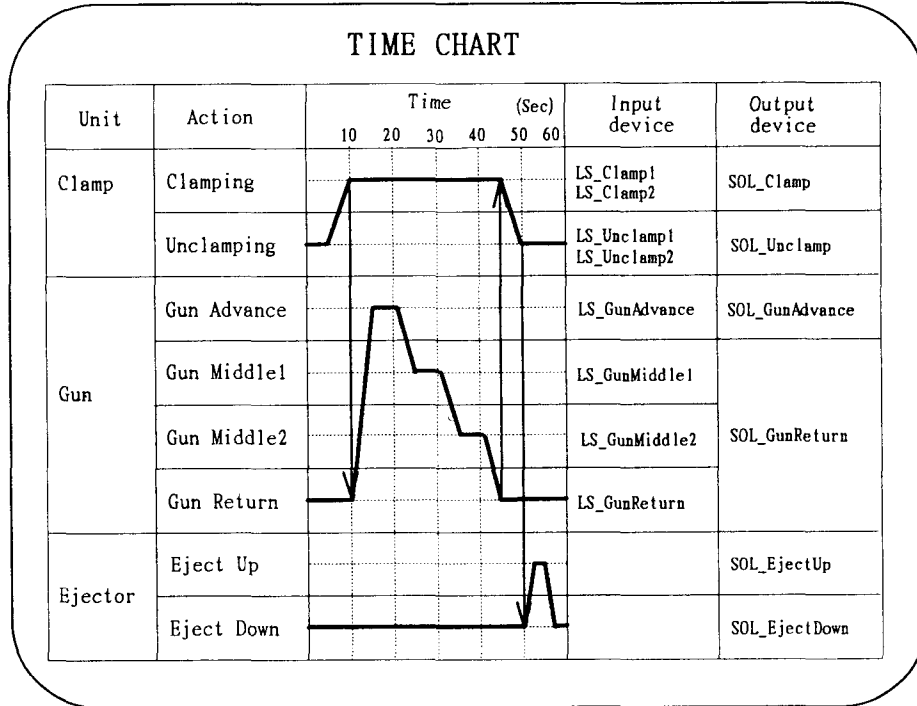


Fig.1 An example of time chart for assembly cell

4. 사례기반추론에 의한 PLC 프로그래밍

PLC 프로그래밍 전문가는 자신이 터득한 지식 및 경험을 바탕으로 새로운 프로그램을 작성하고 관리 하며 기종이 바뀌더라도 과거의 프로그래밍과정에서 체득한 지식을 바탕으로하여 각 기종별 프로그램을 작성하게 된다. 이러한 전문가의 문제해결과정을 그대로 따라가는 컴퓨터시스템이 있다면 PLC프로그래밍과 관련된 전문지식은 그대로 유지되면서 프로그램 표현의 표준화 및 전산화를 이룩할 수 있으며 이는 PLC 프로그래밍의 작업효율을 증대시킬 수 있다. 본 연구를 통해 개발된 프로토타입 지식기반 시스템의 추론방식으로 사례기반추론(CBR: case-based reasoning)을 적용하였다. 이는 현장의 프로그래머가 새로운 PLC 프로그램을 작성할 때, 과거에 작성하였던 래더다이아그램을 참조하면서 수정해 나가는 방법을 적용한 것이다. 사례기반추론은 과거에 경험했던 내용을 사례베이스(case base)에 저장해 놓고 새로운 문제를 접하게 되면 과거의 사례로부터 가장 유사한 사례를 추출하고 이를 적절하게 적용(adaptation)시켜서 주어진 문제를 해결하는 방식이다[6]. 다음의 Fig.2 은 사례기반추론의 일반적인 흐름도이다.

4-1. 사례(case)의 지식표현

전술된 사례기반추론기법으로 문제를 풀기 위해서는 규칙단위가 아닌 사례(case)로서의 지식표현이 필요하다. 각 사례는 문제(problem)와 해(solution)에 관한 정보를 갖게 되고 이를 서술하는데 필요한 속성(property) 들을 객체(object)를 이용하여 형태로 지식표현이 가능하다. 본 연구에서는 타임차트에 표현되는 내용을 문제서술부로 이용하고 타임차트에 해당하는 래더다이아그램 자체를 해부분으로 하는 사례 객체(case object)로 구성된 지식베이스(knowledge base)를 구축하였다.

문제서술부는 타임차트에 관한 속성을 기술하는데 조립셀의 구성요소의 갯수, 구성요소의 이름, 구성요소의 동작, 각 동작간의 순서, 동작을 유발하는 입출력기기의 명칭등이 해당된다.

사례의 해부분에는 래더다이어그램의 각 링의 특징형태(feature), 입력조건부의 점점별 정보, 출력부정보가 표현되어 진다.

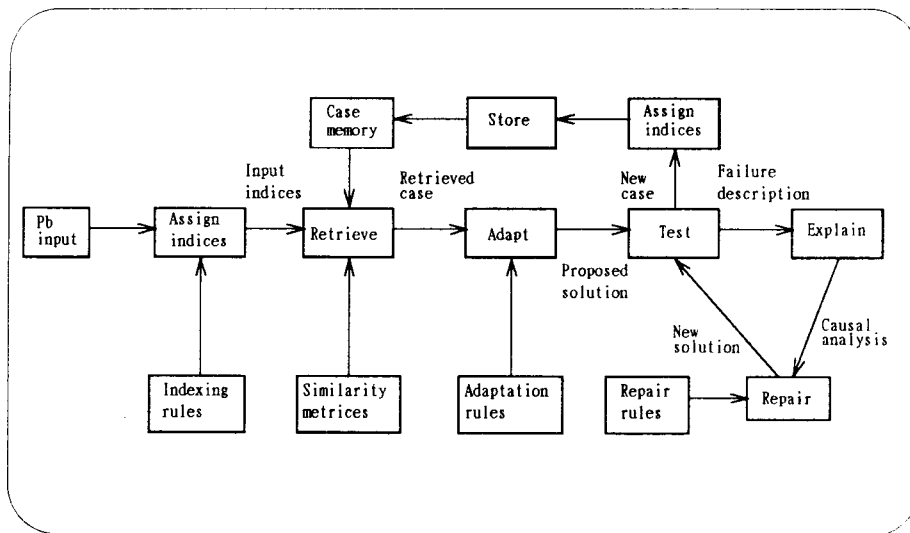


Fig.2 Decision flow of case-based reasoning

#### 4-2. 사례베이스(case base)

사례베이스는 지식베이스(knowledge base)로서 사례의 체계적인 저장을 통하여 효과적인 탐색(search)이 가능하다. 본 연구에서는 각 타임차트의 특징을 작업셀을 구성하는 구동장치별 기준으로 분류되어 이루어 진다. 각 사례에 표현된 구동장치의 종류에 따라 단계적으로 탐색되도록 네트워크(network)을 이루고 있다. 그리고 사례기반추론의 한과정인 적응 과정에서 추론하는데 필요한 PLC 프로그래밍 지식인 구성요소계층도, 제어요소계층도, 구성요소의 동작계층도, 링출력부 제어요소계층도, 링입력부점점별 셋/리셋조건 계층도 등이 또한 사례베이스에 함께 저장되어 있다. Fig.3 은 개발된 시스템의 조립셀의 구성요소 계층도이다.

#### 4-3. 사례의 추출(retrieval)

사례의 추출은 주어진 문제와 저장된 사례중 가장 유사한 사례를 추출하는 작업이며, 이를 위해서는 유사척도(similarity metric)가 필요하다. 본 시스템에서는 구성요소의 갯수와 발생 동작을 중심으로 유사성이 결정되도록 하였다.

#### 4-4. 사례의 적응과 해(solution)의 생성

가장 유사한 사례가 추출되면 이 사례의 해를 이용하여 주어진 문제의 해를 제시해야 한다. 추출사례가 입력문제와 동일한 문제를 갖는다면 사례의 해가 곧바로 주어진 문제의 해가 되며, 상이점이 존재하면 추출된 해를 바탕으로하여 주어진 문제에 맞도록 그 해를 수정한다. 수정되는 내용은 링의 입력부에 관한 셋, 리셋, 인터록, 점검신호등과 출력부의 내부레지스터,타이머, 카운터등에 관한 추가,변경 및 삭제이다. 문제의 적응행위는 규칙으로 표현된 적응휴리스틱스(adaptation heuristics)를 이용하는데, 적응 내용이 복잡하고 많을 수록 결과를 얻기까지 적용되는 규칙들이 많아진다.

본 연구에서는 규칙의 적용횟수 및 적용과정(step)수를 누산하여 이 수치가 일정값(threshold) 이상이 되면 이 문제를 새로운 사례로 저장하도록 하여 학습(learning)이 이루어지도록 설계하였다. 최종적으로 얻어지는 해는 입출력어드레스를 갖지않는 중간래더로직에 관한 텍스트정보이다. 중간래더에 관한 표현 예는 Fig. 4 와 같다.

Fig. 4는 Fig. 1과 같은 타임차트의 정보중 건의 전진(Gun advance) 및 후진(Gun return)을 제어하는 부분만 정리한 것이다. 이때 입력되는 타임차트의 정보는 셀 구동요소의 종류 및 동작을 각각 객체를 정의하여 표현하고 동작간의 순서, 각동작의 개시 및 완료 시점, 동작과 관련된 입출력기기에 대한 정보는 각 객체의 속성값으로 사용자가 입력시켜주게 된다.

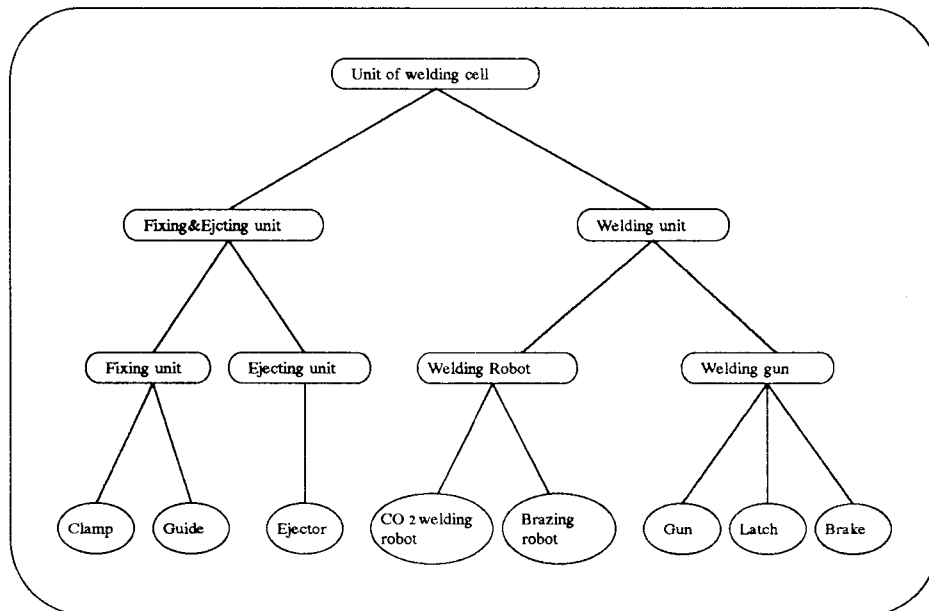


Fig.3 Hierarchy of the components of assembly cell

### 5. PLC 입출력 어드레스 정의와 PLC 프로그램의 생성

출력된 중간래더 다이어그램에 표현된 각 접점들의 리스트를 표의 형태로 표시하여 주면 사용자는 각 접점의 입출력 어드레스를 입력한다. Table 1 은 Fig.4 와 같은 NLD에 표현된 접점들에 대한 리스트의 일부를 보여주고 있다. 시스템은 입출력 어드레스를 입력하도록 표를 제시하게되고 사용자가 해당 어드레스를 입력하여 전체 표를 완성한다.

사용자에 의해 입출력 어드레스가 입력되면 시스템은 최종의 PLC 래더 다이어그램을 생성한다. Fig.5 는 Fig.4 의 중간래더 다이어그램에 Table 1.의 입출력 어드레스가 적용되어 변환된 래더 다이어그램이다.

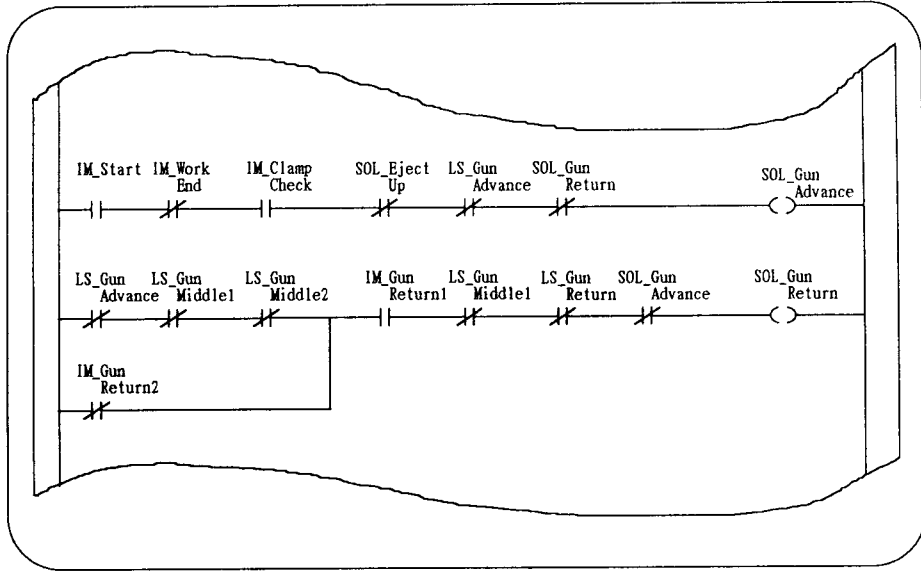


Fig.4 Output neutral ladder diagram

Table 1. I/O address table

I/O Address Table			
Input Module		Output Module	
Device name	Address	Device name	Address
LS_Clamp1	X020	SOL_Clamp	Y030
LS_clamp2	X021	SOL_Unclamp	Y031
LS_Unclamp1	X022	SOL_GunAdvance	Y050
LS_Unclamp2	X023	SOL_GunReturn	Y051
LS_GunAdvance	X040	SOL_EjectUp	Y032
LS_GunMiddle1	X042	SOL_EjectDown	Y033
LS_GunMiddle2	X043		
LS_GunReturn	X044		

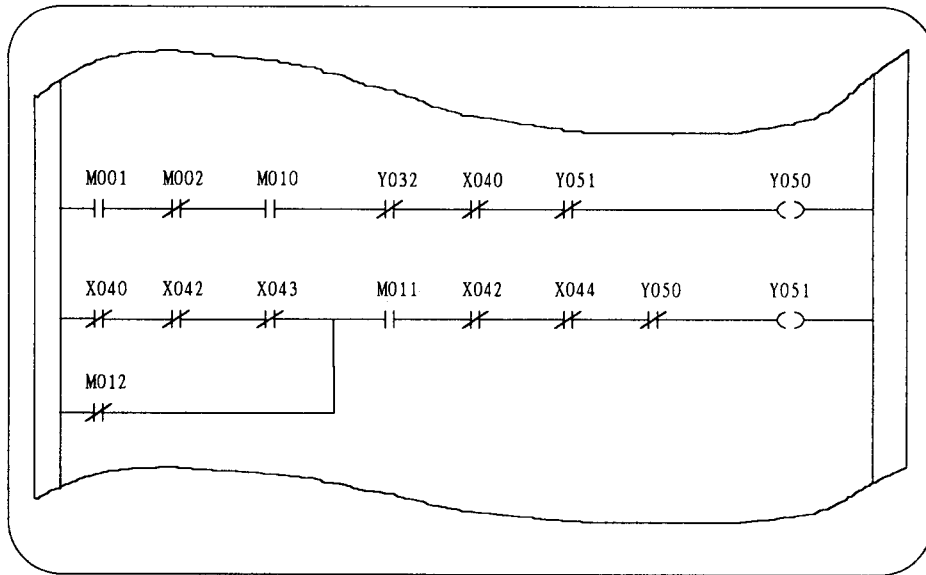


Fig. 5 Converted PLC ladder diagram

## 6. 결론

이상의 기능을 갖는 프로토타입 지식기반 시스템은 IBM 486 호환기종에서 개발되었으며 Nexpert Object[8]와 MS-C 를 이용하여 개발되었다. 시스템의 입력정보인 타임차트정보는 화일로 저장되어 차 후에 비슷한 셀(cell)에 대한 제어프로그램 작성시 간단한 편집과정으로 쉽게 재활용 할 수 있다. 중간래더 다이어그램에 대한 입출력어드레스를 지정하는 과정에서 해당 어드레스를 타기종에 맞는 형태로 지정 하면 동일한 중간래더 다이어그램으로부터 서로 다른 기종의 PLC 래더 다이어그램을 얻을 수 있다. 이는 제어프로그램의 작성시 업무부하를 감소시킬 수 있을 것으로 생각된다. 실업무 적용 분야로는 자동차공장의 신차종 개발에 따른 조립공정에 대한 제어프로그래밍에 이용될 경우 제조준비시간의 단축이 기대된다. 이상과 같이 본 논문에서는 조립셀의 순차제어에 사용되는 프로그래머블(programmable) 기기인 PLC의 프로그램 생성을 자동화 및 지능화하기 위하여 개발된 지식기반 시스템에 관한 기능, 구조 그리고 사례기반추론기법에 관하여 기술하였다. 개발된 시스템은 프로토타입시스템으로 다음과 같은 특징을 갖는다.

- 1) 타임차트정보를 입력으로 하여 중간래더 다이어그램을 자동으로 생성한다.
- 2) 자동생성된 중간래더 다이어그램의 점점에 대해 사용자에게 의해 지정되는 입출력어드레스의 형식에 따라 다양한 기종별로 별도의 래더 다이어그램을 얻을 수 있다.
- 3) 타임차트정보를 입력정보로 하고 있으므로 래더로직과 셀 구동요소의 종류 및 각 동작의 순서에 관련된 정보를 함께 저장, 관리할 수 있다.
- 4) 조립라인의 변동이 생기는 경우 기존 타임차트의 간단한 수정만으로 새로운 래더 다이어그램을 용이하게 얻을 수 있다.
- 5) 제어 프로그램 작성에 따른 시간단축으로 인한 생산성 향상이 기대된다.

이상과 같은 연구에 이어 타임차트 자체를 화면상에서 직접 그래픽으로 입력처리하고 이로부터 속성치를 자동추출하는 시스템, 입출력 어드레스의 지정을 PLC모델만 지정하면 자동으로 처리할 수 있는 시스템으로의 확장연구가 고려되어야 할 것이다.

## 참 고 문 헌

- [1] 青木正夫, "FA 컨트롤러로서의 최근의 PC", *기계설계기술*, 2월호, pp.2-10, 1988.
- [2] Krogh,B.H. and Ekberg,G., "Automatic Programming of Controllers for Discrete Manufacturing Processes", *10th World Congress on Automatic Control Preprints*, International Federation of Automatic Control, Vol.4, pp.160-164, 1987.[4] Devanathan,R., "Computer Aided Design of Relay Ladder Diagram from Functional Specifications", *IECON '90 16th Annual Conference of IEEE Industrial Electronics Society*, pp.527-537, 1990.
- [3] Devanathan,R., "Computer Aided Design of Relay Ladder Diagram from Functional Specifications", *IECON '90 16th Annual Conference of IEEE Industrial Electronics Society*, pp.527-537, 1990.
- [4] Gunasena, N.U. and Lehtihet, E.A., "Automatic Program Generation for PC-Based Sequence Control Problems", *Journal of Manufacturing Systems*, Vol.10 No.2 pp.109-120, 1991.
- [5] Mizutani,H., Nakayama,Y., Ito,S., Namioka,Y. and Matsudaira,T., "Automatic Programming for Sequence Control", *Innovative Application of Artificial Intelligence 4 - Proceedings of the IAAI-92 Conference*, AAAI, pp.315-331, 1992.
- [6] Slade, S., "Case-Based Reasoning: A Research Paradigm", *AI Magazine*, Spring, AAAI, pp.42-55, 1991.
- [7] Warnock, I.G., *Programmable Controllers Operation and Application*, Prentice Hall, pp.346-350, 1992.
- [8] *Nexpert Object Reference Manual*, Neuron Data, 1991.