

SFC로 記述된 매니지먼트 스텝에 의한 개선된 인터록의 실현

(Improved Implementation of Interlock Using Management Step Described by SFC)

유정봉*

(Jeong-Bong You)

요 약

PLC는 산업용 제어시스템에서 가장 많이 사용되며, 중요한 역할을 하고 있다. 이러한 PLC에서 사용되는 언어에 대한 연구도 활발히 이루어지고 있다. PLC에서는 LD언어가 가장 많이 사용되고 있다. LD언어는 데이터의 흐름을 알 수 없고, 유지보수가 어렵다. 반면에 SFC는 제어의 흐름을 이해하기 쉽고 유지보수가 용이하지만 조건과 인터록 기술에 문제점이 있다. 본 논문에서는 매니지먼트 스텝을 사용한 인터록의 기술에 대한 방법을 제안하였고, 실례를 통해 그의 타당성을 확인하였다.

Abstract

Programmable Logic Controller(PLC) is the most widely utilized and plays an important role in industrial control system. The study about a PLC language is performed actively. Ladder Diagram(LD) is the most widely used in PLC. LD is not suitable for describing a complex sequential logic and is very difficult to grasp the sequential flow of control logic and has the disadvantage for a maintenance. On the other hand, Sequential Function Chart(SFC) is very easy to grasp the sequential flow of control logic and has the compatibility for a maintenance but has the disadvantage for describing a condition and a interlock logic. In this paper, we propose the method that describe the interlock logic using management step, and confirm his feasibility through a actual examples.

Key Words : PLC, Ladder diagram, SFC, Interlock, Management Step

1. 서 론

PLC(Programmable Logic Controller)는 최근 다양한 형태 및 복잡한 규모의 제어 시스템에서 많이

사용되고 있다. 또한 릴레이 제어 로직에 비해 유지보수가 용이하고, 기능을 변경하기가 쉬우며, 신뢰성이 높아 산업용 제어장치로서 중요한 역할을 하고 있으며, 그 역할 또한 점점 넓어지고 있다[1].

이러한 PLC는 IEC-1131-3의 국제 규격에 제시된 표준 언어를 사용하게 된다. 이 표준 언어는 IL(Instruction List), ST(Structured Text)의 텍스트 기반언어와 LD(Ladder Diagram), FBD(Fun-

* 주저자 : 공주대학교 전기전자공학부 조교수
Tel : 041-550-0169, Fax : 041-563-3689

E-mail : jbyou@kongju.ac.kr

접수일자 : 2005년 4월 6일

1차심사 : 2005년 4월 15일

심사완료 : 2005년 4월 27일

ction Block Diagram), SFC (Sequential Function Chart)의 그래픽 기반 언어가 있다[2].

이중 LD 언어는 PLC에서 가장 많이 사용되는 표준 언어로 릴레이 제어반의 여러 요소를 소프트웨어로 대체시켜 접점 및 코일로 표현할 수 있게 한 언어이며, 조건과 인터록 논리를 기술하는 데는 상당한 장점을 가지고 있다. 그러나 LD 언어는 숙련된 기술자가 현장경험에 상당히 의존하게 되고, 유지보수를 하기에는 많은 어려움이 있다. 반면에 SFC는 순차 제어 논리에 적합하고, 제어의 흐름을 이해하기가 쉬우며, 에러가 발생했을 때 에러를 찾기가 쉬워 유지보수가 용이하다. 그러나 SFC는 많은 장점이 있지만 조건과 인터록을 처리하기에는 취약한 단점을 가지고 있어 제어 시스템 설계에서 많이 사용되지 못하고 있다[3-5].

PLC를 사용한 산업 제어 시스템에서 LD언어를 주로 사용하지만 비교적 소형이며, 제어 기능이 단순한 제어 시스템에서는 최근에 SFC 언어를 사용하여 설계하고 있다.

SFC 언어를 사용한 제어 시스템 설계를 가능하게 하기 위해서는 SFC 언어의 단점인 조건과 인터록 논리의 기술을 용이하게 해야 할 것이다.

본 논문에서는 매니지먼트 스텝을 사용하여 SFC 언어의 단점인 조건과 인터록 논리의 기술 및 실현을 가능하게 하였으며, 25[inch] TFT-LCD 제조용 In-Line Spin Coater에 적용하여 SFC언어에 의한 인터록 논리의 기술 및 실현이 가능함을 확인하였다.

2. 기존의 방법에 의한 인터록의 記述 및 실현

인터록이란 하나의 동작을 수행하는데 있어서의 잠금장치이며, 동작을 수행하기 위한 기본 조건이다. LD 언어에서 인터록 논리를 기술할 때에는, 인터록 블록은 전체 스캔 프로그램의 앞부분 또는 뒷부분에 일정한 블록에 설정되어, 일련의 동작들에 대한 조건으로 사용된다. 또한 SFC 언어에서 인터록 논리를 기술할 때는 각각의 스텝에 해당하는 인터록 논리들을 모두 부가하는 방법이 있다. 이와 같은 방법을 사용하면, 기술상 불편함이 있고, 많은 용량의 프

로그램 메모리가 필요하게 된다. SFC 언어에서 사용되는 두 번째 방법은 액션 제한자 (N, S, R, L, D, P, SD, DS, SL 등) 중에서 'S(Set 명령)' 제한자를 사용하는 방법이 있다. 이 'S' 제한자를 사용하면, 인터록의 논리 기능 보다는 초기 조건으로써 처음 스캔한 조건만 가능하고, 이후에 바뀌는 조건에 대해서는 의미를 갖지 못하게 된다.

그리고 Hi-Flow 프로그램을 이용하는 방법이 있다. Takao Kokuboi[6]는 인터록 논리를 LD 언어로 프로그램 한 후 인터록 논리들을 통신으로 접수하여 사용하였다. 이와 같은 방법은 통신 레지스터 처리 시간이 길어질 뿐만 아니라, 통신 레지스터 운용 프로그램을 작성해야 하기 때문에 복잡한 프로세스 시스템 설계에는 적합하지 않다.

또한 매크로 스텝을 이용한 방법도 있다. Jeong-Bong You[7]은 END를 검사하지 않는 매크로 스텝을 이용하였다. 이 방법은 매크로 블록을 강제로 무한 루프로 만들고, 매크로 스텝의 다음 천이조건을 만족 시키면, 매크로 블록 내부에 있는 접점의 논리가 바뀌어도 이 접점을 사용하는 그 이후의 스텝에 이 접점의 논리가 그대로 반영되도록 한 것이다. 그러나 이 방법 역시 매크로 블록을 처리할 수 있는 PLC의 종류에 따라서 달라진다. 매크로 블록을 처리할 수 없는 PLC에서는 이와 같은 방법을 사용할 수 없게 된다.

3. 매니지먼트 스텝에 의한 제안된 인터록 논리의 記述 및 실현

매니지먼트 스텝은 시퀀스의 초기 기동시 Set되어 항상 시퀀스의 인터록 블록을 절환 할 수 있고, 흐름을 감시, 제어할 수 있도록 설정한 스텝이다. 각 스텝의 세부내용은 그림 1과 같다.

그림 1의 S1 스텝에서 SW1이 On되면 TR1이 On 된다. 그리고 t1에서 TR1이 On 되기 때문에 trans 조건을 만족하여 S2 스텝으로 전개되고 S2 스텝의 인터록 블록1이 활성화 되어 각각의 인터록 조건에 따라 작동하게 된다. 그리고 endl을 만나면 TR1이 Off 되고 t2 조건을 만족하면서 S3 스텝으로 진행하게 된다. S1 스텝의 SW1이 Off가 되고 SW2가 On

되면 TR2가 On되어 S3 스텝의 인터록 블록2가 활성화되어 각각의 인터록 조건에 따라 작동하게 된다.

따라서 TR1이 On이면 S1의 스텝이 활성화되고, TR2가 On이면 S2의 스텝이 활성화되며, TR3가 On이면 S3의 스텝이 활성화되게 된다. 따라서 각각의 인터록 블록을 독립적으로 처리할 수 있게 된다.

매니지먼트 스텝을 적용한 SFC의 예는 그림 2와 같다.

그림 2의 매니지먼트 스텝은 'S(Set)' 제한자를 갖기 때문에 PLC의 전원이 On되는 순간부터 활성화되어 계속 활성화 상태를 유지하고, 전체 시퀀스의 제어를 가능하게 할 수 있다. Aux 스텝은 매니지먼트 스텝과는 달리 'N(Non-Stored)' 제한자를 사용하며, 해당 스텝이 활성화되었을 때만 활성화 상태가 되고 매니지먼트 스텝에서 하지 않는 부수적인 동작을 제어할 수 있는 보조 역할을 수행한다.

4. 적용 예

4.1 시스템 개요

최근 초고속 정보화의 시대에 발맞추어 디스플레이 산업은 급속한 성장을 보이고 있고, 이러한 추세는 앞으로 상당기간 지속될 것으로 예측된다. 이중 PDP 및 LCD의 기술은 세계의 선두를 유지하고 있다. 그러나 디스플레이에 대한 장치산업은 30%의 국산화율에 머무르고 있는 실정이다. 본 연구에 사용된 시스템은 TFT-LCD 생산공정 중 일부 장비로써 그림 3과 같이 구성되고 있다.

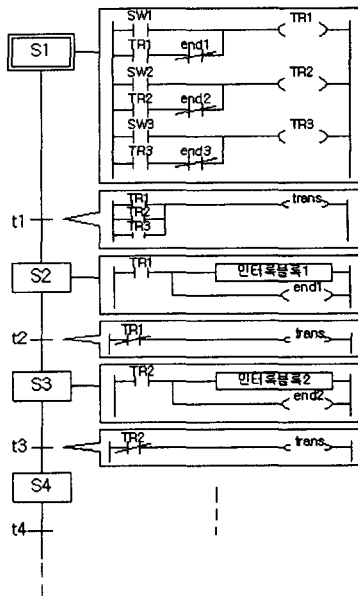


그림 1. 각 스텝과 천이조건の内容
Fig. 1. Description of Each Step and Each transition

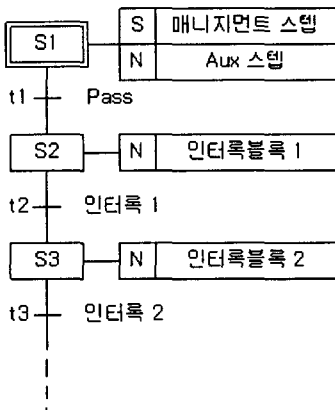


그림 2. 매니지먼트 스텝을 적용한 SFC
Fig. 2. SFC Applied to Management Step

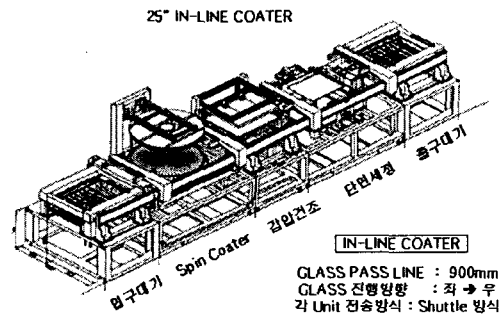


그림 3. In-Line Spin Coater 구성도
Fig. 3. Configuration of In-Line Spin Coater

그림 3의 In-Line Spin Coater는 입구대기부, 스펀코터부, 감압건조부, 단면세정부, 출구대기부 그리고 셔틀부의 6개 유닛으로 구성된다. In-Line Spin Coater는 유리기판에 포토레지스트 도포액을 얇게 도포하는 장비이다. 입구대기부는 포토레지스트를 도포하기 위해 이송하는 유닛이며, 스펀코터부는 포토레지스트가 도포된 기판을 회전시켜 포토레지스트

SFC로 記述된 매니지먼트 스텝에 의한 개선된 인터록의 실현

트를 기관 전체에 균일하게 분산 도포시켜주는 유닛이다. 그리고 감압건조부는 스펀코터부의 진공 척과 단면세정부의 진공 척의 온도차이로 인한 유리기관 이면에 흔적이 발생하는 것을 방지하기 위한 유닛이다. 또한 레지스트가 도포된 것을 세정하는 단면세정부와 기관을 다른 공정으로 이동시켜주는 출구대기부가 있다.

4.2 알고리즘 설계

전체 알고리즘을 설계하기 위하여 각 6개의 유닛을 개별 설계한 후 서틀에 동기시켰다. 본 알고리즘은 공동 프로그램과 서틀의 이동 동작, 그리고 각 유닛의 고유작업으로 나누어진다. 공통 프로그램은 전원투입후의 원점복귀부, 기구부의 위치판정부 그리고 인터록 블록으로 구성된다.

본 논문 실험을 통해 SFC 언어에서 인터록의 기술이 정확히 실현되는지를 설명하기 위해 간단히 전체 알고리즘 중에서 입구대기부의 알고리즘만을 나탄낸다. 입구대기부의 알고리즘은 그림 4와 같다.

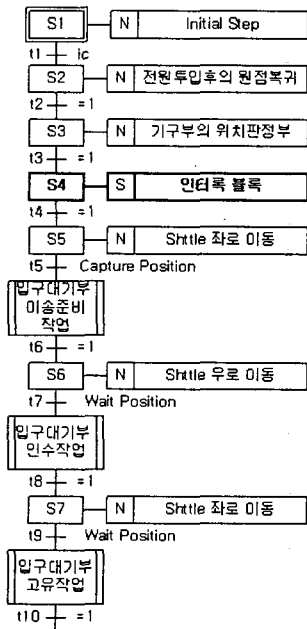


그림 4. 입구대기부 알고리즘
Fig. 4. Algorithm of Input Unit

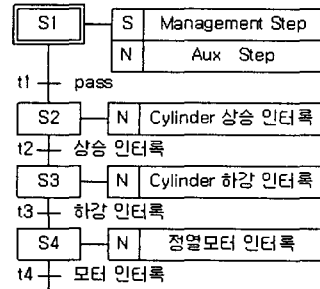
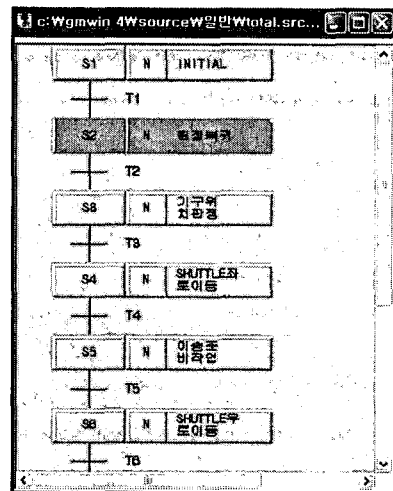


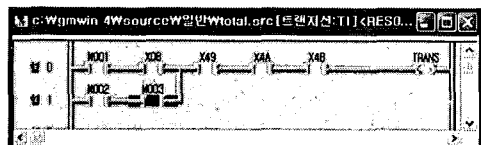
그림 5. 입구대기부 인터록 블록
Fig. 5. Interlock Block of Input Unit

그림 4의 입구대기부의 알고리즘에서 4번 스텝에 인터록 블록을 설정하였으며, 이 인터록 블록은 다시 매니지먼트 스텝을 이용하여 각 실린더에 대한 개별 인터록을 설정하여 인터록 접점을 검사하고 실행하게 하였다. 이 입구대기부의 인터록 블록은 그림 5와 같다.

4.3 실험결과 및 검토



(a)SFC

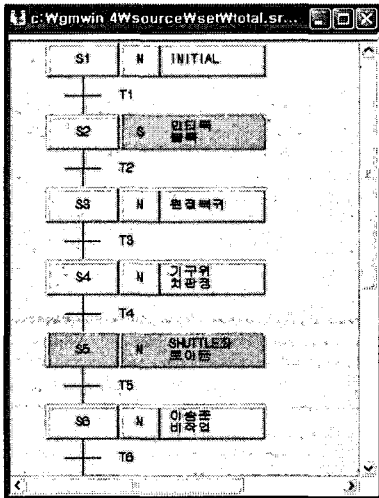


(b)T1의 내용

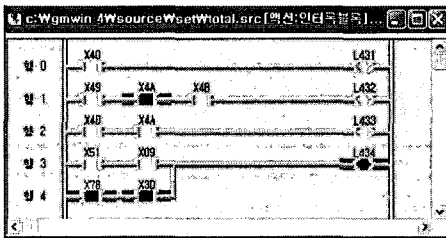
그림 6. 일반적인 알고리즘의 적용 결과
Fig. 6. Results applied by general Algorithm

본 연구를 위해서 PLC는 LG 산전의 GLOFA GM4기종의 CPU를 사용하였으며, 편집 프로그램은 GMWIN Ver 4.0을 사용하였다[8-9].

그림 6에서는 일반적으로 SFC언어에서 사용하는 방법을 보여준다. 별도의 인터록 블록을 설정하지 않고 각 트랜지션에 조건과 인터록 점점들을 모두 삽입하는 것으로 코딩시 불편할 뿐만 아니라 메모리 용량이 상당히 커지게 된다.



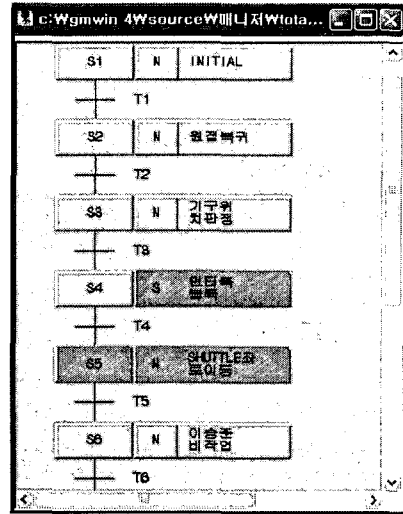
(a)SFC



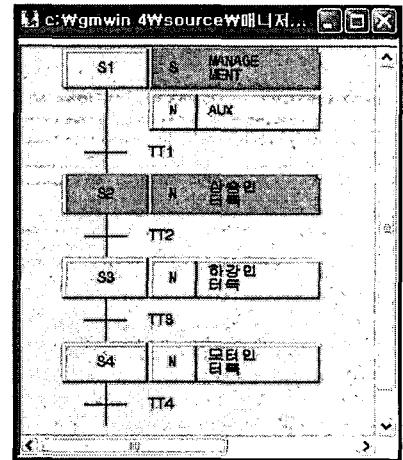
(b)인터록 블록의 내용

그림 7. SET 제한자 적용 결과
Fig. 7. Results applied by SET Instruction

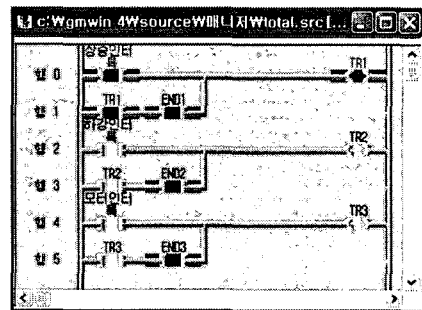
또 다른 방법으로 'S(set)'제한자를 사용하는 방법으로 그림 7과 같다. 그림 7(a)에서 S2 스텝이 'S'제한자를 사용한 인터록 블록으로 설정되어 있다. 본 SFC는 S5 스텝이 활성화되어 있어도 인터록 블록도 'S'제한자 때문에 계속 활성화되어 있다. 그림 7(b)에서 L434 인터록 점점이 On 되어 있는 것을 알 수 있다.



(a)입구대기부 알고리즘



(b)인터록블록



(c)매니지먼트 스텝

그림 8. 입구대기부의 적용 결과
Fig. 8. Results Applied to Input Unit

SFC로 記述된 매니지먼트 스텝에 의한 개선된 인터록의 실현

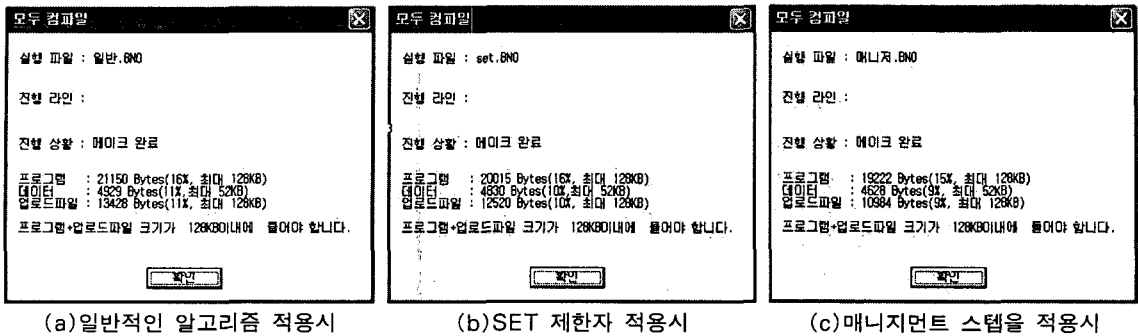


그림 9. 메모리 크기 비교
Fig. 9. Comparison of Memory Size

이것은 S2 스텝이 활성화로 되어 한 번 set되면, 인터록 조건이 변경되어도 그 이후의 활성화스텝에 영향을 주지 못한다. 이러한 이유로 'S' 제한자를 사용했을 때는 인터록으로써의 제 기능을 발휘하지 못한다.

본 논문에서 제안한 매니지먼트 스텝을 사용한 방법은 그림 8과 같다.

그림 8(a)에서 S4스텝이 'S' 제한자를 사용한 인터록 블록으로 설정되어있다. 이 SFC는 S5 스텝이 활성화되어 있어도 'S' 제한자를 사용한 S4 스텝은 계속 활성화되어 있음을 알 수 있다. 또한 이 인터록 블록은 다시 SFC로 기술하였으며 그림 8(b)와 같다.

그림 8(b)에서 S1 스텝이 'S' 제한자를 사용한 매니지먼트 스텝으로 S2 스텝이 활성화되어 있지만 S1의 매니지먼트 스텝도 동시에 활성화 되어 있다. 또한 그림 8(c)는 매니지먼트 스텝의 LD 내용을 보여주고 있다. 'TR1'출력이 On 되어 있는 것을 볼 수 있다.

이 것은 그림 8(b)의 S2 스텝을 활성화시키는 접점이다. 따라서 매니지먼트 스텝을 적용하면 각각의 인터록 블록을 독립적으로 사용할 수 있다는 것을 그림 8을 통해서 알 수 있다. 또한 위의 세 가지 예에 대해 메모리의 양을 비교한 것이 그림 9와 같다.그림 9에서 (a) 일반적인 알고리즘을 적용했을 때 프로그램 메모리 크기는 21150B이고, (b) 'S' 제한자를 사용했을 때는 20015B, 본 논문에서 제안한 (c) 매니지먼트 스텝을 적용했을 때는 19222B이다. 즉 매니지먼트 스텝을 적용했을 때는 프로그램 메모리의 크기가 일반적인 알고리즘을 적용했을 때보다 약 10[%]

정도 줄어들었음을 알 수 있다.

5. 결론

산업 제어 시스템에서 제어기로서 PLC를 많이 사용하고 있으며, PLC의 표준언어로 LD 언어가 대부분 사용되고 있다. LD 언어는 데이터를 처리하고 유지보수 하는데 단점이 있어 현장 경험이 풍부한 숙련된 기술자들이 주로 사용해왔다. 반면에 SFC는 순차 제어 논리에 적합하고, 제어의 흐름을 이해하기가 쉬우며, 유지보수가 용이하여 프로그램을 작성하고 운용하는데 상당히 편리하다.

그러나 SFC는 인터록을 처리하는데 많은 문제점을 가지고 있었다. 본 논문에서 매니지먼트 스텝을 사용하여 SFC 언어의 취약점인 인터록 논리에 대한 기술 및 실현 방법을 제안하였고, 적용 예를 통해 그의 타당성을 확인하였다. 매니지먼트 스텝을 사용하였을 때가 기존의 방법에 비해 우수함이 입증되었습니다.

본 연구를 통해, 매니지먼트 스텝을 사용하면 SFC 언어로 프로그램 작성시 PLC의 종류에 무관하게 인터록을 처리할 수 있으며, 각 유닛의 개별 인터록 블록으로 설정하여 독립적 인터록 처리가 가능한 것으로 판단이 되었다. 앞으로 산업 제어 시스템에서 SFC 언어를 사용하여 효율적이고 용이한 설계가 가능할 것으로 기대된다.

References

- [1] Bong-Suk Kang and Kwang-Hjun Cho, "Discrete Event Model Conversion Algorithm for Systematic Analysis of Ladder Diagrams in PLCs" *Journal of Control, Automation and systems Engineering*, Vol 8. No5, p401-406, May, 2002.
- [2] R.W.Lewis, "Programming Industrial Control Systems Using IEC1131-3", The Institution of Electrical Engineers, 1992.
- [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 Transactions on Mechatronics*, Vol 1, No4, pp290-294, Dec 1996.
- [4] T Kouthon M. A. Peraldiabd J. D. Deco-tignie "Distributing PLC Control", *International Conf. on IEC. IEEE 21'st Vol 2*. pp1614-1619. 1995.
- [5] M. Zhou and E Twiss, "Design of Industrial automated systems via relay ladder logic programming and Petrinets", *IEEE Trans on Systems, Man and Cybernetics-part C : Applications and Reviews*, Vol 28, No 1, pp 137- 150, 1998.
- [6] Takao Kokubo, Kenzo Kamiyama, Masaharu Oku and Hitoshi Saito. "Application of Powerful SFC language(Hi-Flow) To FA Systems in a Tireindustry ", 16th annual Conference of IEEE IES. 1990.
- [7] Jeong-Bong You, Kwang-Jun Woo, Kyung-Moo Hyu, "Implementation of Interlock in Process Control System Described by Sequential Function Chart Graphical Language", *조명·전기설비학회 논문지*, Vol. 12, No. 2, May 1998.
- [8] "Mitsubishi PLC Programming Manual", Mitsubishi, OnA series, 2004.
- [9] "LG Programmable Logic Controller Glofa-GM", LG Industrial Systems, 2004.

◇ 저자소개 ◇

유정봉 (庾正鳳)

1964년 3월 5일생. 1988년 2월 단국대학교 전자공학과 졸업. 1990년 8월 동대학원 전자공학과 졸업(석사). 1998년 8월 동대학원 전자공학과 졸업(박사). 1990년 7월~1993년 9월 (주)신도리코. 현재 공주대학교 전기전자공학부 조교수. 주요관심분야는 PLC 제어, 마이크로프로세서 제어, BLDC 모터 제어, 디스플레이 장비 공장자동화 알고리즘 설계 등.