이산사건모델에 기반한 PLC 래더다이어그램 자동합성

Synthesis of Ladder Diagrams for PLCs Based on Discrete Event Models

강 봉 석, 조 광 현
(Bong-Suk Kang and Kwang-Hyun Cho)

Abstract : PLC(Programmable Logic Controller)s are essential components of modern automation systems encompassing almost every industry. Ladder Diagrams (LD) have been widely used in the design of such PLC since the LD is suitable for the modeling of the sequential control system. However, the synthesis of LD itself mainly depends on the experience of the industrial engineer, which may results in unstructured or inflexible design. Hence, in this paper, we propose a ladder diagram conversion algorithm which systematically produces LDs for PLCs based on discrete event models to enhance the structured and flexible design mechanism.

Keywords : discrete event model, state diagram, ladder diagram, PLC, boolean equation

I. 서론

공장자동화에 필수적인 기기인 PLC(Programmable Logic Controller)는 입력, 출력, 기억, 연산, 제어부를 갖추어 컴퓨터와 유사한 기능을 수행하는 시퀀스 제어장치로서 사용자의 프로그램에 의해 운전되기 때문에 릴레이 로직에 비해 유연성, 보수성, 신뢰성이 월등히 높아 대다수의 산업용 기기 제어장치로 사용되고 있다. 이러한 PLC의 제어언어로는 LD(Ladder diagram), IL(Instruction List), SFC(Sequential Function Chart), FBD (Function Block Diagram), ST (Structured Text) 등이 주로 사용되고 있다[1]-[3]. 이 가운데 현장 사용자에게 가장 익숙하고 그래픽 제어언어인 래더다이어그램(LD)을 보편적으로 사용하고 있다. 그런데 래더다이어그램을 작성하기 위해서는 사용자가 시스템의 순차제어 방법을 정하고, 각 공정의 동시성과 비동기적인 성질을 고려하여 작성해야 한다. 따라서 이러한 기존의 래더다이어그램 작성방법은 숙련된 프로그래머의 축적된 현장 경험과 제안된 알고리즘에 크게 의존하여 여러 시행착오를 거쳐 이루어지므로 불필요한 인력과 많은 시간을 소요해야 할 뿐만 아니라 작성된 래더다이어그램의 수정을 위한 유연성 등이 결여되어 있는 실정이다.

한편, 유연성이 높고 사용자가 편리하게 래더다이어그램을 작성하기 위한 많은 연구가 진행중이며 대표적인 방법이 이산사건모델(discrete event model)을 이용한 방법이다. 대표적인 이산사건모델로는 페트리네트(Petri net)와 상태선도(state diagram 또는 automata) 등이 있다. 래더다이어그램에 의한 시퀀스 제어를 페트리네트 기반의 제어기로 대체하고자 하는 연구는 1980년대부터 연구되기 시작하여 현재까지 다방면으로 시도되어져 왔다[4]-[7]. 그러나 지금까지 이루어진 페트리네트를 이용한 래더다이어그램 변환에 관한 연구는 생성된 래더다이어그램이 일반적이지 못할 뿐 아니라 길어지고 복잡하여 실제 시스템에 적용시키기에는 아직도 많은 어려움이 따른다. 본 논문에서는 이산사건모델 중에서 상태선도를 이용한 래더다이어그램 변환 알고리즘을 제안한다. 이러한 상태선도는 직관적이고 체계적으로 시스템을 파악하고 분석할 수 있을 뿐 아니라 모델링 측면에서도 우수한 성능을 지니고 있다. 따라서, 상태선도로 대상시스템의 동적특성을 모델링하고 이를 입력받아 래더다이어그램으로 변환하는 알고리즘을 제안함으로써 체계적이고 유연하며 실제 적용가능한 래더다이어그램을 설계 하고자 한다. 단, 본 논문에서 제안하는 알고리즘은 대상시스템의 논리적 동작사양을 만족시키는 PLC의 래더다이어그램 자동합성을 목적으로 한다.

본 논문의 구성은 다음과 같다. 2절에서는 시스템을 모델링하기 위한 상태선도의 사건(event) 및 상태(state)를 정의한다. 3절에서는 상태선도를 이용하여 블리언 방정식(Boolean equation)으로 유도하고 래더다이어그램으로 변환하는 알고리즘을 기술한다. 4절에서는 컨베이어 시스템을 상태선도로 모델링한 예제를 이용하여 래더다이어그램으로 변환한 후 모의실험을 통하여 결과를 검증한다. 5절에서는 결론을 맺고 향후 연구과제를 논의한다.

II. 시스템 모델링

래더다이어그램은 현장의 작업 상황을 순차적으로 작성한 심볼들의 집합으로서 단지 순차적 표현으로는 적합하지만 시스템 모델링이나 분석 측면에서는 한계를 지니고 있다. 따라서 대상 시스템을 직관적으로 파악하고 체계적으로 시스템을 해석하고 분석하기 위하여 이산사건모델을 적용하고자 한다. 이산사건모델의 논리적인 동작을 다루는 모델 중 가장 대표적인 방법들 중 하나는 상태선도에 기반한 방법이다. 상태선도는 시스템의 상태가 변화하는 순서, 각 상태들의 조건 그리고 상태천이의 조건 등을 도식화한 것이다. 각 상태는 원으로 표시하고 상태사이의 천이는 원을 연결하는 화살표로 표시한다. 본 논문에서는 상태선도를 래더다이어그램으로 변환하는데 있어서 PLC의 동작특성을 고려하여 사건 및 상태를 다음과 같이 정의한다:

접수일자 : 2001. 1. 19., 수정완료 : 2001. 6. 18.
강봉석, 조광현 : 울산대학교 전기전자정보시스템공학부(kbs@sys.ulsan.ac.kr/ckh@mail.ulsan.ac.kr)
※ 본 연구는 2000년도 울산대학교 연구비와 2001년도 한국학술진흥재단의 연구비(KRF-2001-041-E00279)에 의하여 수행되었습니다.

1) 상태는 통상 PLC의 내부출력, 외부출력, 타이머출력, 카운터출력 등으로 정의한다.

2) 내부출력, 외부출력, 타이머출력, 카운터 출력 각각의 상태는 동작을 위한 셋(set) 상태와 정지를 위한 리셋(reset) 상태로 구분한다.

3) ε 은 시스템의 내부적인 변화에 의한 상태천이를 나타낸다.

III. 래더다이어그램 변환 알고리즘

본 절에서는 상태선도를 이용하여 래더다이어그램으로 변환하는 알고리즘을 소개한다. 먼저 상태선도를 이용하여 블리언 방정식을 유도한 후 래더다이어그램으로 변환한다. 그림 1은 위에서 정의한 상태 및 사건을 이용하여 시스템의 동작특성을 표현한 상태선도로부터 래더다이어그램으로 변환 생성하는 과정을 개략적으로 표현한 것이다. 그림 1에서 보는 바와 같이 래더다이어그램 변환 알고리즘은 2개의 주요 단계로 구성된다.

그림 1의 정의에 따른 상태선도로부터 래더다이어그램으로 변환하기 위한 알고리즘은 다음과 같다. 먼저 단계 1은 상태선도로부터 블리언 방정식을 유도하는 과정을 수도코드(pseudo code)로 나타낸 것이다. 여기서, 상태선도로부터 블리언 방정식을 유도하기 위한 상태(Q)와 사건(E)에 대하여 다음과 같이 정의한다: α_1, α_2, …, α_{i-1}, α_i, α_{i+1}와 같은 상태들($\alpha_i \in Q$)이 사건의 천이에 의하여 순서대로 정의되어 있다고 가정하면 $d(\alpha_i, \alpha_{i+1}) = \alpha_i$이다. 여기서, α_i는 입력상태이고, α_{i+1}은 출력상태를 나타낸다. 그리고 $E = e(\alpha_i, \alpha_{i+1})$는 입력상태 αi에서 출력상태 αi+1로 천이하는 사건을 나타낸다(단, $e(\alpha_{i-1}, \alpha_i) = e(\alpha_i, \alpha_{i+1}) = \varepsilon$이면 $d(\alpha_{i-1}, \alpha_{i+1}) = \alpha_{i-1}$이다). 즉, α_{i-1}, α_i, α_{i+1}의 상태가 사건에 의하여 순차적으로 정의되어 있고, 각각의 상태에서 ε사건이 연속으로 발생했을 때 처음으로 ε 사건이 발생한 상태를 입력상태로 정의된다. 또한, $g : Q \mapsto \{0, 1\}$와 $f : Q \mapsto \{0, 1\}$는 각각 다음과 같이 정의된다: $f(\alpha_j)$는 $\delta(\alpha_j, E) = \alpha_i$일 때 '1'이고, 그 외의 경우는 '0'이다. 여기서, δ는 천이함수로서 상태 α_j로부터 사건 E에 의하여 상태 α_i로의 천이를 나타낸다. 그리고 $g(x\,\alpha_j)$는 $\alpha_j \in Y_{j,s}$이고, $\alpha_i \in Y_{i,s}$이거나, $i \neq j$인 경우 $\alpha_j \in Y_{j,s}$이면서 $\alpha_i \in Y_{i,r}$이거나, $\alpha_j \in Y_j$이고, $\alpha_i \in X_i$이면, $g(\alpha_j) = f(\alpha_j)$이고, 그 외의 경우는 '1'이다. 이때 첨자 j는 입력 상태(단, $1 \le j \le N$), 첨자 i는 출력상태(단, $1 \le i \le N$), 첨자 s는 셋(set) 상태, 첨자 r은 리셋(reset) 상태를 각각 나타낸다. 또한, 변수 N은 상태의 수, X는 타이머 또는 카운터 상태, Y는 내부출력 또는 외부출력 상태를 각각 나타낸다. 내부출력 또는 외부출력 상태의 셋(set) 상태 $Y_{i,s}$가 출력상태일 때 자기 자신의 상태를 입력상태로 갖는다고 가정한다.

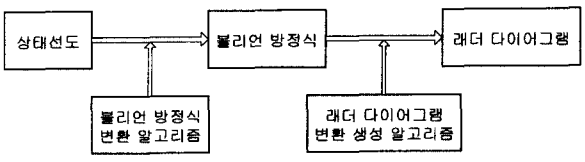

그림 1. 래더다이어그램 변환 블록다이어그램.
Fig. 1. Block diagram of the proposed conversion procedures.

단계 1 : 상태선도로부터 블리언 방정식으로의 변환과정

begin
for ($i = 1$; $i \le N$; $i++$)
  if (*output state* $== Y_{i,s}$)
    for ($j = 1$; $j \le N$; $j++$)
      if (*input state* $== Y_{j,s}$ and $i == j$)
        then $Y_{i,s}[j] = Y_{j,s}$
      else if (*input state* $== Y_{j,s}$ and $i \neq j$)
        then $Y_{i,s}[j] = g(d(Y_{j,s}, Y_{i,s})) \cdot E$
      else $g(d(Y_{j,r}$ or $X_j, Y_{i,s})) = 1$ and $Y_{i,s}[j] = E$
      end if

else if (*output state* $== Y_{i,r}$)
  for ($j = 1$; $j \le N$; $j++$)
    if (*input state* $== Y_{j,s}$ and $i == j$)
      then $g(d(Y_{j,s}, Y_{i,r})) = 1$ and $Y_{i,r}[j] = E$
    else if (*input state* $== Y_{j,s}$ and $i \neq j$)
      then $Y_{i,r}[j] = g(d(Y_{j,s}, Y_{i,r})) \cdot E$
    else $g(d(X_i, Y_{i,r})) = 1$ and $Y_{i,r}[j] = E$
    end if
  $Y_{i,s} = \sum_{j=1}^{N} Y_{i,s}[j]$, $\quad Y_{i,r} = \sum_{j=1}^{N} Y_{i,r}[j]$
  $Y_i = Y_{i,s} \cdot \overline{Y_{i,r}}$

else if (*output state* $== X_{i,s}$)
  for ($j = 1$; $j \le N$; $j++$)
    if (*input state* $== Y_j$)
      then $X_{i,s}[j] = g(d(Y_j, X_{i,s})) \cdot E$
    else $g(d(X_j, X_{i,s})) = 1$ and $X_{i,s}[j] = E$
    end if
else if (*output state* $== X_{i,r}$)
  for ($j = 1$; $j \le N$; $j++$)
    if (*input state* $== X_j$)
      then $g(d(X_j, X_{i,r})) = 1$ and $X_{i,r}[j] = E$
    else $X_{i,r}[j] = g(d(Y_j, X_{i,r})) \cdot E$
    end if
  $X_{i,s} = \sum_{j=1}^{N} X_{i,s}[j]$, $\quad X_{i,r} = \sum_{j=1}^{N} X_{i,r}[j]$
end if
end

위의 수도코드는 PLC를 제어하기 위한 동작사양을 이용하여 상태선도로 모델링되었을 때 이를 블리언 방정식으로 유도하는 과정을 나타낸 것이다. 래더다이어그램을 생성하기 위한 블리언 방정식은 상태선도로 모델링된 각각의 상태에 대하여 적용하게 된다. 또한 블리언 방정식은 각각의 상태로 입력하는 입력상태와 사건에 의하여 유도할 수 있다. 즉, 출력상태가 내부출력 또는 외부출력상태의 셋(set) 상태 $Y_{i,s}$일 때 $Y_{i,s}$로 입력되는 모든 입력상태에 대하여 입력상태가 자기 자신의 상태일 때 $Y_{i,s}$[j]에는 자기 자신의 상태를 저장하고, 자기 자신의 상태를 제외한 $Y_{j,s}$가 입력상태일 때 $Y_{i,s}$[j]에는 입력상태와 사건의 논리곱으로 표현되며, 그 밖의 입력상태에 대해서는 사건만을 저장한다. 이와 같은 과정을 거친 후 $Y_{i,s}$[j]의 모든 값에 대하여 논리합 연산을 통해 $Y_{i,s}$에 저장된다. 또한, 출력상태가 내부출력 또는 외부출력상태의 리셋(reset) 상태 $Y_{i,r}$일 때에도 비슷한 과정에 의하여 블리언 방정식이 유도된다. 이렇게 유도된 각각의 상태에 대하여 $Y_i = Y_{i,s} \cdot \overline{Y_{i,r}}$을 이용하여 하나의 출력으로 내보내지게 된다. 출력상태가 타이머 또는 카운터 상태의 셋(set) 상태 $X_{i,s}$일 때 $X_{i,s}$로 입력되는 모든 입력상태에 대하여 입력상태가 $Y_{j,s}$일 때 $X_{i,s}$[j]에는 입력상태와 사건이 함께 저장되며, 그 밖의 입력상태에 대해서는 사건만을 저장한다. 이와

같은 과정을 거친 후 $X_{i,s}$[j]의 모든 값에 대하여 논리합 연산을 통해 Xi,s에 저장된다. 또한, 출력상태가 타이머 또는 카운터출력상태의 리셋(reset) 상태 $X_{i,r}$도 비슷한 과정에 의하여 블리언 방정식이 유도된다.

다음 단계 2는 단계 1에서 유도된 블리언 방정식을 이용하여 래더다이어그램으로 변환하는 과정을 나타낸 것이다. 상태선도에서 래더다이어그램으로의 변환에 사용된 기호는 표 1에 기술하였다. 여기서 P_j는 PLC에서의 입력접점을 나타내며 상태선도에서의 사건을 나타낸다.

표 1. 상태선도에서 래더다이어그램으로의 변환표.
Table 1. Conversion table of state diagrams into ladder diagrams.

상태선도	래더다이어그램	설 명
P_j	P_j	상시개로접점
$\overline{P}_j$	$\overline{P}_j$	상시폐로접점
M_i	M_i	내부출력
P_i	P_i	외부출력
T_{is}	TON T_i k	온딜레이 타이머 k는 설정시간
T_{ir}	RST T_i	타이머 리셋
$C_{i,s}$ $C_{i,r}$	U CTU C_i R <S> k	업 카운터 k는 설정값

단계 2 :

1. $f(Y_{j,s})$와 $f(Y_{j,r})$을 Y_j로 치환,
2. 논리곱은 직렬로, 논리합은 병렬로 연결,
3. Y_j와 P_j는 a접점으로, $\overline{Y}_j$와 $\overline{P}_j$는 b접점으로 치환,
4. 각각의 출력상태를 내부출력, 외부출력, 타이머, 카운터 등으로 치환.

그림 2는 상태선도가 주어졌을 때 이를 이용하여 래더다이어그램으로 변환하기 위하여 각각의 출력에 대하여 입력상태와 사건을 이용하여 블리언 방정식을 유도하고, 이를 PLC의 제어 프로그램인 래더다이어램으로 변환하기 위하여 입력접점, 타이머, 카운터, 내부출력, 외부출력 등으로 치환하는 과정을 나타낸다. 단계 1에서 유도된 블리언 방정식에서 $f(Y_{j,s})$와 $f(Y_{j,r})$을 Y_j로 치환한 후 논리곱은 래더다이어그램에서 직렬로 연결되고 논리합은 병렬로 연결된다. 또한, 블리언 방정식에서의 Y_j와 P_j는 래더다이어그램에서 상시개로접점인 a접점으로, $\overline{Y}_j$와 $\overline{P}_j$는 상시폐로접점인 b접점으로 치환한 후 각각의 출력상태를 래더다이어그램에서의 내부출력, 외부출력, 카운터, 타이머 등으로 치환하면 래더다이어그램이 생성된다.

IV. 시스템 적용사례

본 절에서는 위에서 제시한 래더다이어그램 변환 알고리즘을 컨베이어 시스템에 적용하여 각 단계를 예제를 통해 기술한다.

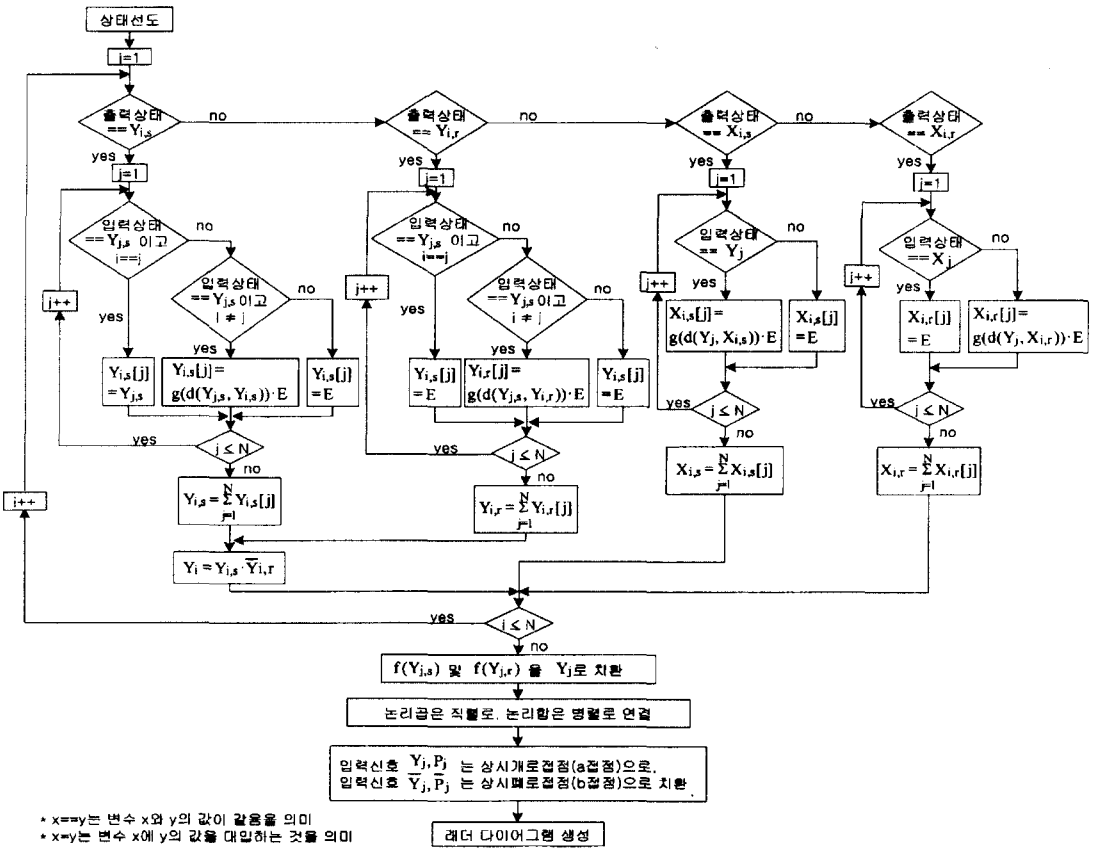

그림 2. 래더다이어그램 변환 플로우차트.
Fig. 2. Flow chart of the conversion procedures.

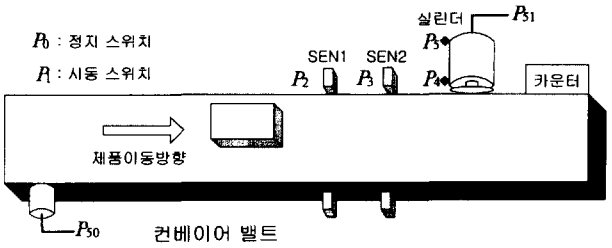

그림 3. 컨베이어 시스템 구성도.
Fig. 3. The conveyor system.

그림 3은 시동 스위치(P_1)을 눌렀을 때 컨베이어 기동용 모터(P_{50})가 동작하게 되고 컨베이어 시스템에 제품이 들어오면 컨베이어 밸트를 따라 제품이 이동하고 하한검출센서 SEN1(P_2)에서 물체를 검출하면 타이머가 동작한다. 제품이 계속 이동하여 상한검출센서 SEN2(P_3)에서 물체를 검출하면 상한검출센서의 내부출력(M_2)이 동작하여 하한검출센서의 내부출력(M_1)을 정지시키고 타이머를 정지시키며, 카운터를 이용하여 제품의 수량을 개수하고 'P_3=0'이 되면 상한검출센서의 내부출력이 정지한다. 한편, P_3에서 물체검출에 실패하면 불량품으로 간주하여 타이머 설정시간인 5초 후 실린더가 동작(P_{51})하여 제품을 밖으로 배출시키는 공장의 일부분을 나타낸 것이다.

그림 4는 컨베이어 시스템의 동작특성을 고려하여 이산사건모델로 표현한 그림이다. 여기서, $P_{50,r}$은 컨베이어 시스템의 초기상태를 나타내며, 두 줄의 원으로 표시된 $M_{2,r}$, $P_{51,r}$ 그리고 $C_{1,r}$은 컨베이어 시스템의 최종상태를 나타낸다. 그림 4를 이용하여 블리언 방정식을 유도하는 과정을 설명하면 다음과 같다. 첫째, 그림 4의 상태선도로부터 컨베이어 기동용 모터 P_{50}의 셋(set) 상태는$g(d(P_{50,r}, P_{50,s}))=1$, $g(d(P_{50,s}, P_{50,s}))=f(P_{50,s})$이므로 $P_{50,s}= P_1 \cdot g(P_{50,r})+g(P_{50,s})=P_1+f(P_{50,s})$이 되고, P_{50}의 리셋(reset) 상태는 $g(d(P_{50,s}, P_{50,r}))=1$이므로 $P_{50,r}= g(P_{50,s}) \cdot P_0=P_0$가 된다. 따라서 $P_{50}=P_{50,s} \cdot \overline{P}_{50,r}= (P_1+f(P_{50,s})) \cdot \overline{P}_0$로 표현된다. 둘째, 싱글 실린더 밸브 구동을 위한 P_{51}의 셋(set) 상태는 $g(d(T_{1,s}, P_{51,s}))=1$, $g(d(P_{51,s}, P_{51,s}))=f(P_{51,s})$이므로 $P_{51,s}=g(T_{1,s}) \cdot T_1 \cdot P_5+g(P_{51,s})=T_1 \cdot P_5+f(P_{51,s})$이 되고, P_{51}의 리셋(reset) 상태는 $g(d(T_{1,r}, P_{51,r}))=1$이므로 $P_{51,r}=g(T_{1,r}) \cdot P_4=P_4$가 된다. 그러므로 $P_{51}=P_{51,s} \cdot \overline{P}_{51,r}=(T_1 \cdot P_5+f(P_{51,s})) \cdot \overline{P}_4$와 같은 블리언 방정식을 얻을 수 있다. 셋째, 하한검출센서 SEN1의 내부출력상태 M_1의 셋(set) 상태는 $g(d(P_{50,s}, M_{1,s}))=f(P_{50,s})$, $g(d(M_{1,s}, M_{1,s}))=f(M_{1,s})$이므로 $M_{1,s}=f(P_{50,s}) \cdot P_2+f(M_{1,s})$가 되고, M_1의 리셋(reset) 상태는 $g(d(M_{2,s}, M_{1,r}))+f(M_{2,s})$,

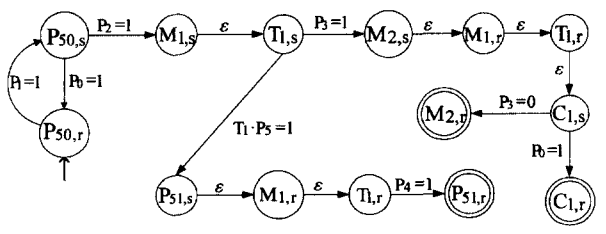

그림 4. 컨베이어 시스템의 상태선도.
Fig. 4. State diagram of the conveyor system.

표 2. 컨베이어 시스템의 변수정의.
Table 2. Definition of the variables for the conveyor system.

구분	PLC I/O	입출력에 대한 설명
입력	P_0	정지 스위치
	P_1	시동 스위치
	P_2	제품의 하한검출센서
	P_3	제품의 상한검출센서
	P_4	제품의 전진검출 리미트 스위치
	P_5	제품의 후진검출 리미트 스위치
내부출력	M_1	제품의 하한검출 내부출력
	M_2	제품의 상한검출 내부출력
카운터	C_1	업 카운터
타이머	T_1	온딜레이 타이머
외부출력	P_{50}	컨베이어 기동용 모터
	P_{51}	싱글 실린더 밸브 구동

$g(d(P_{51,s}, M_{1,r}))=f(P_{51,s})$이므로 $M_{1,r}=f(M_{2,s})+f(P_{51,s})$이 된다. 따라서 $M_1=M_{1,s} \cdot \overline{M}_{1,r}=(f(P_{50,s}) \cdot P_2+f(M_{1,s})) \cdot \overline{f(M_{2,s})} \cdot \overline{f(P_{51,s})}$이 된다. 넷째, 상한검출센서 SEN2의 내부출력상태 M_2의 셋(set) 상태는 $g(d(T_{1,s}, M_{2,s}))=1$, $g(d(M_{2,s}, M_{2,s}))=f(M_{2,s})$이므로 $M_{2,s}=g(T_{1,s}) \cdot P_3+g(M_{2,s})=P_3+f(M_{2,s})$와 같이 되고, M2의 리셋(reset) 상태는 $g(d(C_{1,s}, M_{2,r}))=1$이므로 $M_{2,r}=g(C_{1,s}) \cdot \overline{P}_3=\overline{P}_3$이 된다. 따라서 $M_2=M_{2,s} \cdot \overline{M}_{2,r}=(P_3+f(M_{2,s})) \cdot \overline{\overline{P}}_3=(P_3+f(M_{2,s})) \cdot P_3$가 된다.

다섯째, 온딜레이 타이머 T_1의 셋(set) 상태는 $g(d(M_{1,s}, T_{1,s}))=f(M_{1,s})$이므로 $T_{1,s}=f(M_{1,s})$가 되고, T_1의 리셋(reset) 상태는 $e(M_{2,s}, M_{1,r})=e(M_{1,r}, T_{1,r})=\varepsilon$이므로 $g(d(M_{2,s}, T_{1,r}))=f(M_{2,s})$이고, $e(P_{51,s}, M_{1,r})=e(M_{1,r}, T_{1,r})=\varepsilon$이므로 $g(d(P_{51,s}, T_{1,r}))=f(P_{51,s})$가 되어 $T_{1,r}=f(M_{2,s})+f(P_{51,s})$이 된다. 마지막으로, 업 카운터 C_1의 셋(set) 상태는 $e(M_{2,s}, M_{1,r})=e(M_{1,r}, T_{1,r})=e(T_{1,r}, C_{1,s})=\varepsilon$이므로 $g(d(M_{2,s}, C_{1,s}))=f(M_{2,s})$가 되어 $C_{1,s}=f(M_{2,s})$가 되고, C_1의 리셋(reset) 상태는 $g(d(C_{1,s}, C_{1,r}))=1$이므로 $C_{1,r}=g(C_{1,s}) \cdot P_0=P_0$와 같은 블리언 방정식을 얻을 수 있다. 이상에서 얻어진 블리언 방정식으로부터 $f(Y_{j,s})$와 $f(Y_{j,r})$을 Y_j로 치환한 후 논리곱은 직렬로 연결하고, 논리합은 병렬로 연결한다. 또한, 래더다이어그램에서 입력신호는 왼쪽에 배치하고 출력상태는 오른쪽에 배치한다. Y_j, P_j는 a접점으로 변환하고, $\overline{Y}_j$와 $\overline{P}_j$는 b접점으로 변환하면 최종적으로 래더다이어그램이 생성된다. 그림 5는 제시된 알고리즘에 의해 유도된 래더다이어그램이다. 유도된 래더다이어그램의 각 순서대로 동작실행 추이를 검증하여 보면 주어진 논리적 동작사양이 모두 만족됨을 확인할 수 있다. 이를테면, 그림 5의 1번 라인의 경우 시동 스위치 P_1은 상시개로접점이므로 P_1을 눌렀을 때 P_1의 블리언 방정식은 '1'이 되고, 정지 스위치 P_0는 상시폐로접점이므로 P_0를 누르지 않았을 때 P_0의 블리언 방정식은 '1'이

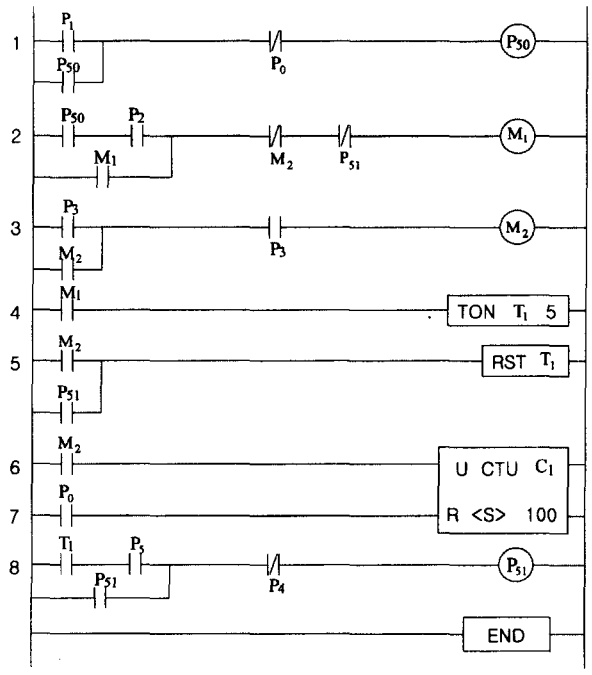

그림 5. 컨베이어 시스템의 래더다이어그램.
Fig. 5. Ladder diagram of the conveyor system.

되어 컨베이어 기동용 모터 P_{50}가 동작함을 알 수 있다. 마찬가지로 여타의 동작조건에 대한 추이도 이와 같은 방식으로 검증하여 주어진 논리적 동작사양을 모두 만족함을 알 수 있다.

V. 결론

본 논문에서는 래더다이어그램에 대한 전문지식이 없어도 시스템 동작사양만을 파악하고 있으면 래더다이어그램을 작성할 수 있도록 이산사건모델을 이용하여 PLC의 제어 알고리즘인 래더다이어그램으로의 변환기법을 제안하였다. 또한 래더다이어그램 변환 알고리즘을 컨베이어 시스템에 적용하였다. 향후 연구과제로는 제안된 래더다이어그램 생성과정의 효율화 및 PLC의 고급기능의 명령어들을 포함시키는 연구 등이 있다.

참고문헌

[1] E. A. Parr, *Programmable Controllers, An Engineer's Guide*, 3rd ed. Oxford, U. K.: Newnes, 1999.

[2] A. Falcione and B. H. Krogh, "Design recovery for relay ladder logic," *IEEE Control Systems Magazine*, vol. 13, no. 2, pp. 90-98, 1993.

[3] R. W. Lewis, *Programming industrial control systems using IEC 1131-3*, 2nd ed. London, U. K.: The Institution of Electrical Engineers, 1995.

[4] M. Uzam, A. H. Jones, and N. Ajlouni, "Conversion of Petri net controllers for manufacturing systems into ladder logic diagrams," *Proc. IEEE Conf. on Emerging Technologies and Factory Automation*, Kauai, Hawaii, vol. 2, pp. 649-655, 1996.

[5] M. Uzam and A. Jones, "Discrete event control system design using automation Petri nets and their ladder diagram implementation," *International Journal of Advanced Manufacturing Technology*, vol. 14, no. 10, pp. 716-728, 1998.

[6] M. Zhou and E. Twiss, "Design of Industrial automated systems via relay ladder logic programming and Petri nets," *IEEE Trnas. on Systems, Man, and Cybernetics-part C: Applications and Reviews*, vol. 28, no. 1, pp. 137-150, 1998.

[7] K. Feldmann and A. W. Colombo, "Specification, design, and implementation of logic controllers based on colored Petri net models and the standard IEC 1131 Part I: Specification and design," *IEEE Trans. on Control Systems Technology*, vol. 7, no. 6, pp. 657-674, 1999.

강 봉 석

1975년생. 2000년 울산대학교 전기공학과 졸업. 2000년~현재 울산대학교 전기전자정보시스템공학부(舊 전자공학과) 석사과정 재학중. 관심분야는 PLC 응용기술 및 생산시스템 자동화, 통신네트워크에서의 라우팅 및 혼잡제어, 이산사건시스템의 관리제어 등.

조 광 현

1993년 한국과학기술원 전기 및 전자공학과 졸업. 동대학원 석사(1995). 동대학원 박사(1998). 한국과학기술원 위촉연구원(1998-1999). 1999-현재 울산대학교 전기전자정보시스템공학부(舊 전자공학과) 조교수. 관심분야는 이산사건시스템의 관리제어, 생산시스템 자동화, 통신망 분석 및 제어, 광통신망, 지능형 교통관제시스템, 바이오정보공학 등.