



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년07월10일
(11) 등록번호 10-2132659
(24) 등록일자 2020년07월06일

(51) 국제특허분류(Int. Cl.)
G05B 19/418 (2006.01) G06N 3/08 (2006.01)
G06N 99/00 (2019.01)
(52) CPC특허분류
G05B 19/41885 (2013.01)
G06N 20/00 (2019.01)
(21) 출원번호 10-2018-0071559
(22) 출원일자 2018년06월21일
심사청구일자 2018년06월21일
(65) 공개번호 10-2020-0003323
(43) 공개일자 2020년01월09일
(56) 선행기술조사문헌
KR1020160063990 A*
US20090089234 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
포항공과대학교 산학협력단
경상북도 포항시 남구 청암로 77 (지곡동)
(72) 발명자
황형주
경상북도 포항시 남구 지곡로 155, 8동 1501호(지곡동, 교수아파트)
황락훈
대구광역시 북구 동천로 155 102동 1203호 (동천동, 칠곡네스빌)
김진오
부산광역시 부산진구 연수로8번길 39-10(양정동)
(74) 대리인
특허법인이름리온

전체 청구항 수 : 총 5 항

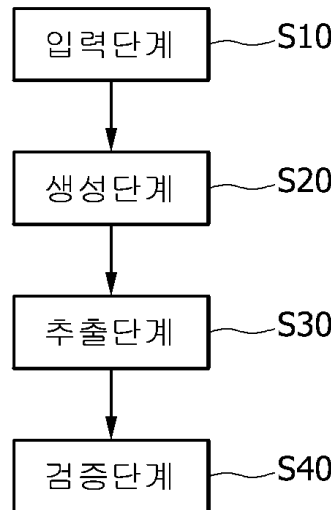
심사관 : 강성철

(54) 발명의 명칭 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법

(57) 요약

본 발명은 인공지능기반 역공학 이용한 공장 자동화용 제어기의 분석 방법에 관한 것으로, a) 분석대상 지능 제어기의 입출력 데이터를 입력받아 전처리하는 입력단계와, b) 전처리된 입출력 데이터들을 인공지능 학습을 통해 논리 모델을 구축하는 단계와, c) 구축된 모델을 통해 분석대상인 지능 제어기의 내부 논리를 추출하는 단계를 포함한다.

대표도 - 도1



(52) CPC특허분류
G06N 3/08 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711069980
부처명	과학기술정보통신부
연구관리전문기관	재단법인한국연구재단
연구사업명	전략과제(산업수학)
연구과제명	딥러닝 알고리즘의 수학적 접근 및 AI 활용 금융기술 개발 연구
기여율	1/1
주관기관	포항공과대학교 산학협력단
연구기간	2018.03.01 ~ 2019.02.28

명세서

청구범위

청구항 1

- a) 분석대상 지능 제어기의 입출력 데이터를 입력받아 전처리하는 단계;
 - b) 전처리된 입출력 데이터들을 인공지능 학습을 통해 논리 모델을 구축하는 단계; 및
 - c) 구축된 모델을 통해 분석대상인 지능 제어기의 내부 논리를 추출하는 단계를 포함하되,
- 상기 a) 단계에서 입출력 데이터는 이산형 또는 연속형이며, 데이터의 형태에 따라 상기 b) 단계의 학습 모델에 차이를 두며,
- 상기 전처리는, 입출력 데이터를 마이크로초 단위로 이산화(Discretization)하여 수열(array) 형태로 처리하는 것을 특징으로 하는 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법.

청구항 2

삭제

청구항 3

제1항에 있어서,
 상기 b) 단계는,
 의사결정 나무 분류기 학습을 수행하는 단계; 및
 모델 정확도가 설정된 기준 이상인지 확인하여 설정된 기준 이상이 아니면 상기 a) 단계를 재수행하고,
 기준 이상이면 c) 단계를 수행하는 단계를 포함하는 인공지능기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법.

청구항 4

제1항에 있어서,
 상기 b) 단계는,
 상기 입출력 데이터가 이산형인 경우 신경망 모델을 이용하여 학습하는 것을 특징으로 하는 인공지능기반 역공학을 이용한 공장 자동화용 제어기의 분석방법.

청구항 5

제1항에 있어서,
 상기 c) 단계는,
 나무 분류기를 이용하여 분석 후 진로결정 과정을 순서도를 작성하고,
 상기 순서도를 기반으로 래더 로직 다이어그램을 작성하여,
 진로결정을 가시화하는 것을 특징으로 하는 인공지능기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법.

청구항 6

제1항에 있어서,
 상기 c) 단계는,
 신경망 모델 또는 선형 회귀 모델을 사용하여 지능 제어 장치의 내부 로직을 추출하는 것을 특징으로 하는 인공

지능기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법.

발명의 설명

기술 분야

[0001] 본 발명은 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법에 관한 것으로, 더 상세하게는 기계 학습 알고리즘을 이용하여 프로그램 가능한 로직 컨트롤러(Programmable Logic Controller, PLC), 직접 계수 제어 장치(Direct Digital Controller, DDC), 분산 처리 시스템(Distributed Control System, DCS), 독립 실행형 컨트롤러(Standalone) 등과 같은 지능 제어 시스템(Intelligent Controller)의 구조 및 작동 원리를 파악하고 이를 재현하기 위한 방법에 관한 것이다.

배경 기술

[0002] 최근 스마트 공장 등 공장 자동화를 위하여 지능 제어기의 도입이 활발하게 이루어지고 있다. 지능 제어기는 프로그램 가능한 로직 컨트롤러(Programmable Logic Controller, PLC), 직접 계수 제어 장치(Direct Digital Controller, DDC), 분산 처리 시스템(Distributed Control System, DCS), 독립 실행형 컨트롤러(Standalone)를 포함하는 개념이다.

[0003] 공장 자동화를 위해서는 위의 제어기들의 유기적인 동작의 결합관계의 파악이 매우 중요하다. 하지만, 현재까지 국내에서는 각종 공정 제어 시스템을 구축할 때 해외 제조 업체로부터 장비를 수입하고, 수입된 설비들의 작동 원리나 유기적이 결합관계를 분석할 수 없기 때문에 설비의 유지 보수에 많은 비용을 지급하고 있다.

[0004] "Abdelhameed, Magdy M., Darabi, Houshang. Diagnosis and debugging of programmable logic controller control programs by neural networks. Proceedings of the 2005 IEEE International Conference on Automation Science and Engineering, Edmonton, Canada, August 1 & 2, 2005, ISBN: 0-7803-9426-7, 313-318 쪽"에는 순환 신경망을 이용하여 프로그램 가능한 로직 컨트롤러를 모델링하고 오류를 검출하거나 진단하는데 필요한 방안을 제시하였다.

[0005] 하지만 알고리즘 학습 과정의 추적이나 분석 과정이 결여되어, 컨트롤러의 내부 작동원리를 규명할 수 없으며, 다양한 알고리즘을 적용하지 않아 모델 정확도를 검증하기 어렵다는 한계가 존재한다.

[0006] "Prähofer, Herbert; Schatz, Roland; Wirth, Christian; Mossenbock, Hanspeter. A Comprehensive Solution for Deterministic Replay Debugging of SoftPLC Applications. IEEE Transactions on Industrial Informatics, Volume 7, Issue 4, November 2011, Article number 6009197, ISSN: 15513203, 641-651쪽", "Prähofer, Herbert; Wirth, Christian; Berger, Richard. Reverse engineering and visualization of the reactive behavior of PLC applications. IEEE International Conference on Industrial Informatics (INDIN), 2013, Article number 6622946, ISBN: 978-1-4799-0752-6, 564-571쪽", "Pirzadeh, Heidar; Agarwal, Akanksha; Hamou-Lhadj, Abdelwahab. An Approach for Detecting Execution Phases of a System for the Purpose of Program Comprehension. 2010 8th ACIS International Conference on Software Engineering Research, Management and Applications, Article number 5489832, ISBN: 978-076954075-7, 207-214쪽" 에서는 프로그램 가능한 로직 컨트롤 시스템의 추적 로그(trace log)를 이용하여 컨트롤러를 모델링하고 오류를 검출하는 방법을 제시하였다.

[0007] 하지만, 이와 같은 종래의 방법은 시스템 내부 작동 원리가 어느 정도 규명이 되어 있는 상태에서 모델링한다는 점에 있어서 큰 제약 조건을 가지고 있으며, 특히 추적 로그 정보를 얻을 수 없는 경우, 적용이 불가능하다는 단점이 있다.

선행기술문헌

비특허문헌

[0008] (비특허문헌 0001) Abdelhameed, Magdy M., Darabi, Houshang. Diagnosis and debugging of programmable logic controller control programs by neural networks. Proceedings of the 2005 IEEE International Conference on Automation Science and Engineering, Edmonton, Canada, August 1 & 2, 2005, ISBN: 0-7803-

9426-7, 313-318쪽

ä(비특허문헌 0002) Prhofer, Herbert; Schatz, Roland; Wirth, Christian; Mossenbock, Hanspeter. A Comprehensive Solution for Deterministic Replay Debugging of SoftPLC Applications. IEEE Transactions on Industrial Informatics, Volume 7, Issue 4, November 2011, Article number 6009197, ISSN: 15513203, 641-651쪽

ä(비특허문헌 0003) Prhofer, Herbert; Wirth, Christian; Berger, Richard. Reverse engineering and visualization of the reactive behavior of PLC applications. IEEE International Conference on Industrial Informatics (INDIN), 2013, Article number 6622946, ISBN: 978-1-4799-0752-6, 564-571쪽

(비특허문헌 0004) Pirzadeh, Heidar; Agarwal, Akanksha; Hamou-Lhadj, Abdelwahab. An Approach for Detecting Execution Phases of a System for the Purpose of Program Comprehension. 2010 8th ACIS International Conference on Software Engineering Research, Management and Applications, Article number 5489832, ISBN: 978-076954075-7, 207-214쪽

발명의 내용

해결하려는 과제

- [0009] 본 발명이 해결하고자 하는 기술적 과제는, 분석대상을 가동하여 입출력 데이터를 확보하고, 이를 기계 학습 알고리즘을 통해 처리하여 의사 결정 모델을 구축함으로써, 제어 시스템 내부 작동 원리를 역으로 파악할 수 있는 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법을 제공함에 있다.
- [0010] 본 발명이 해결하고자 하는 다른 기술적 과제는, 지능 제어기의 출력값이 이산값인지 또는 연속값인지에 따라 분류하고, 각각에 대하여 적당한 알고리즘을 제공하여 정확성이 높은 기계 학습 모델을 구축함으로써, 이후 알고리즘의 학습 과정을 추적하여 지능 제어기의 구조 및 작동원리를 파악하고, 나아가 대체 제어기를 모델링할 수 있는 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법을 제공함에 있다.

과제의 해결 수단

- [0011] 상기와 같은 과제를 해결하기 위한 본 발명 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법은, a) 분석대상 지능 제어기의 입출력 데이터를 입력받아 전처리하는 단계와, b) 전처리된 입출력 데이터들을 인공지능 학습을 통해 논리 모델을 구축하는 단계와, c) 구축된 모델을 통해 분석대상인 지능 제어기의 내부 논리를 추출하는 단계를 포함한다.
- [0012] 본 발명의 일실시예에 따르면, 상기 a) 단계에서 입출력 데이터는 이산형 또는 연속형이며, 데이터의 형태에 따라 상기 b) 단계의 학습 모델에 차이를 두며, 상기 전처리는, 입출력 데이터를 마이크로초 단위로 이산화(Discretization)하여 수열(array) 형태로 처리할 수 있다.
- [0013] 본 발명의 일실시예에 따르면, 상기 b) 단계는, 의사결정 나무 분류기 학습을 수행하는 단계와, 모델 정확도가 설정된 기준 이상인지 확인하여 설정된 기준 이상이 아니면 상기 a) 단계를 재수행하고, 기준 이상이면 c) 단계를 수행하는 단계를 포함할 수 있다.
- [0014] 본 발명의 일실시예에 따르면, 상기 b) 단계는, 상기 입출력 데이터가 이산형인 경우 신경망 모델을 이용하여 학습할 수 있다.
- [0015] 본 발명의 일실시예에 따르면, 상기 c) 단계는, 나무 분류기를 이용하여 분석 후 진로결정 과정을 순서도를 작성하고, 상기 순서도를 기반으로 래더 로직 다이어그램을 작성하여, 진로결정을 가시화할 수 있다.
- [0016] 본 발명의 일실시예에 따르면, 상기 c) 단계는, 신경망 모델 또는 선형 회귀 모델을 사용하여 지능 제어 장치의 내부 로직을 추출할 수 있다.

발명의 효과

- [0017] 본 발명 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법은, 지능 제어기의 입출력 데이터를

이용하여 제어 시스템 내부 작동원리를 파악할 수 있어, 선진 공정 제어기술을 확보할 수 있는 효과가 있다.

[0018] 본 발명은 동일 공정의 플랜트 신규 건설 또는 증설시 지능 제어기들을 국산화할 수 있으며, 유지 및 보수가 용이하도록 할 수 있는 효과가 있다.

도면의 간단한 설명

[0019] 도 1은 본 발명의 바람직한 실시예에 따른 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법의 순서도이다.

도 2는 도 1의 알고리즘 순서도이다.

도 3 내지 도 6은 각각 분석대상인 프로그램 가능한 로직 컨트롤러(PLC)에 입력되는 입력 변수들(P1-P4)의 히스토그램의 예시도이다.

도 7 내지 도 12는 출력변수(ACC, DEC, STD, FORWARD, RUN, STOP)의 히스토그램의 예시도이다.

도 13은 진로 결정과정의 순서도이다.

도 14 내지 도 19는 각각 나무 기반 모델을 분석하여 작성한 출력변수들의 래더 로직 다이어그램의 예시도이다.

도 20은 출력변수 ACC의 모델 생성 과정에서 선택한 신경망 모델의 예시도이다.

도 21은 출력변수 DEC의 모델 생성 과정에서 선택한 신경망 모델의 예시도이다.

도 22는 신경망 모델을 분석하여 작성한 출력변수 DEC의 래더 로직 다이어그램의 예시도이다.

발명을 실시하기 위한 구체적인 내용

[0020] 이하, 본 발명 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법에 대하여 첨부한 도면을 참조하여 상세히 설명한다.

[0021] 본 발명의 실시 예들은 당해 기술 분야에서 통상의 지식을 가진 자에게 본 발명을 더욱 완전하게 설명하기 위해 제공되는 것이며, 아래에 설명되는 실시 예들은 여러 가지 다른 형태로 변형될 수 있으며, 본 발명의 범위가 아래의 실시 예들로 한정되는 것은 아니다. 오히려, 이들 실시 예는 본 발명을 더욱 충실하고 완전하게 하며 당업자에게 본 발명의 사상을 완전하게 전달하기 위하여 제공되는 것이다.

[0022] 본 명세서에서 사용된 용어는 특정 실시 예를 설명하기 위하여 사용되며, 본 발명을 제한하기 위한 것이 아니다. 본 명세서에서 사용된 바와 같이 단수 형태는 문맥상 다른 경우를 분명히 지적하는 것이 아니라면, 복수의 형태를 포함할 수 있다. 또한, 본 명세서에서 사용되는 경우 "포함한다(comprise)" 및/또는 "포함하는(comprising)"은 언급한 형상들, 숫자, 단계, 동작, 부재, 요소 및/또는 이들 그룹의 존재를 특정하는 것이며, 하나 이상의 다른 형상, 숫자, 동작, 부재, 요소 및/또는 그룹들의 존재 또는 부가를 배제하는 것이 아니다. 본 명세서에서 사용된 바와 같이, 용어 "및/또는"은 해당 열거된 항목 중 어느 하나 및 하나 이상의 모든 조합을 포함한다.

[0023] 본 명세서에서 제1, 제2 등의 용어가 다양한 부재, 영역 및/또는 부위들을 설명하기 위하여 사용되지만, 이들 부재, 부품, 영역, 층들 및/또는 부위들은 이들 용어에 의해 한정되지 않음은 자명하다. 이들 용어는 특정 순서나 상하, 또는 우열을 의미하지 않으며, 하나의 부재, 영역 또는 부위를 다른 부재, 영역 또는 부위와 구별하기 위하여만 사용된다. 따라서, 이하 상술할 제1 부재, 영역 또는 부위는 본 발명의 가르침으로부터 벗어나지 않고서도 제2 부재, 영역 또는 부위를 지칭할 수 있다.

[0024] 이하, 본 발명의 실시 예들은 본 발명의 실시 예들을 개략적으로 도시하는 도면들을 참조하여 설명한다. 도면들에 있어서, 예를 들면, 제조 기술 및/또는 공차에 따라, 도시된 형상의 변형들이 예상될 수 있다. 따라서, 본 발명의 실시 예는 본 명세서에 도시된 영역의 특정 형상에 제한된 것으로 해석되어서는 아니 되며, 예를 들면 제조상 초래되는 형상의 변화를 포함하여야 한다.

[0025] 도 1은 본 발명의 바람직한 실시예에 따른 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법의 순서도이고, 도 2는 알고리즘 순서도이다.

[0026] 도 1과 도 2를 각각 참조하면 본 발명의 바람직한 실시예에 따른 인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법은, 입력단계(S10), 생성단계(S20), 추출단계(S30), 분석단계(S40)를 포함한다. 본 발명의

인공지능 기반 역공학을 이용한 공장 자동화용 제어기의 분석 방법은 통상의 연산이 가능한 제어기와 저장장치 및 데이터의 임시 저장이 가능한 메모리를 포함하는 장치를 이용하여 이루어지며, 예를 들어 퍼스널 컴퓨터를 사용할 수 있다.

- [0027] 먼저, 입력단계(S10)는 분석 대상인 지능 제어기에 장착된 센서들로부터 입출력 데이터를 입력 받아 전처리한다.
- [0028] 그 다음, 생성단계(S20)는 입력단계에서 입력된 데이터의 형태에 따라 적절한 예측 모델을 선택한 후, 학습시킨다.
- [0029] 입력된 데이터의 형태는 데이터가 이산값 또는 연속값으로 구분할 수 있다.
- [0030] 추출단계(S30)에서는 생성단계(S20)에서 학습된 모델을 분석하여 지능 제어 장치의 내부 로직을 찾아내는 단계로, 나무 기반 모델, 신경망 모델 또는 선형 회귀 모델을 사용할 수 있다.
- [0031] 그 다음, 검증단계(S40)에서는 추출단계(S30)에서 추출된 로직을 검증한다.
- [0032] 이하에서는 본 발명의 각 단계에 대하여 더욱 구체적으로 설명한다.
- [0033] 먼저, 입력단계(S10)에서는, 상기 지능 제어기의 센서 입출력 데이터를 입력 받아 전처리한다. 이때, 입출력 데이터의 형태에 따라 지능 제어 장치의 논리 분석 방법이 달라지므로 데이터를 전처리한다.
- [0034] 입력단계(S10)에서는 연속적 시계열(continuous time series) 형태의 입출력 데이터를 값의 변동 정보를 잃지 않는 최소 시간 간격, 예를 들어, 마이크로초 단위로 이산화(Discretization)하여 수열(array) 형태로 저장한다.
- [0035] 예측을 하는데 현재 시점의 입력만으로 충분하다면 추가적인 데이터 전처리 과정 없이 이 수열을 그대로 사용한다. 반면, 예측 모델 구축 과정에 과거 시점의 데이터가 필요한 경우, 예를 들어, 시간 지연 모듈이 포함된 경우, 나무 기반 모델을 적용할 때, 얼마나 이전 데이터를 사용할지 결정하고 특정 시간 단계(time step)만큼 입력값들을 잘라서 시간 창(time window)을 만든다.
- [0036] 바로 직전 시점의 출력값들을 현재 시점의 입력값으로 사용하는 로직이 자주 나타나는 경우, 예를 들어, 자기지지(self-holding)형 로직 단위(logic unit)가 포함된 경우 등에서는 직전 출력값들도 분류기(classifier)의 입력값에 함께 포함시킬 수 있다. 고동(pulse) 형식이 아니라 계단 함수(step function) 형식의 출력값의 경우 신호(signal)가 켜지고 꺼지는 순간을 예측하는 것이 더 직관적이므로, 이에 맞게 데이터를 전처리한다.
- [0037] 도 3 내지 도 6은 분석대상인 프로그램 가능한 로직 컨트롤러(PLC)에 입력되는 입력 변수들(P1~P4)의 히스토그램의 예시도이고, 도 7 내지 도 12는 출력변수(ACC, DEC, STD, FORWARD, RUN, STOP)의 히스토그램의 예시도이다.
- [0038] 생성단계(S20)에서는 상기 입력단계(S10)에서 입력된 데이터의 형태가 이산값인 경우, 나무기반 모델(Tree-based model)을 활용하여 학습을 수행한다.
- [0039] 본 발명에서는 시간 간격을 조절해가면서 의사 결정 나무 분류기(decision tree classifier)를 학습시킨다. 예측에 대한 평가지표로 ROC AUC 점수를 사용할 수 있다. 학습 된 트리로부터 분석 대상인 지능 제어기의 로직을 찾아내기 위해서는 나무 의사 결정 모델을 최대한 간단히 만드는 것이 중요하다. 따라서, 입력값으로 사용되는 데이터의 시간 간격은 최소한으로 줄이고, 나무의 구조는 최대한 단순하게 될 수 있도록 모델을 만든다.
- [0040] 모델 구축 과정에서 주의할 점으로, 부적절한 계량 함수(metric)를 평가지표로 사용할 경우, 예측이 완전히 어긋나더라도 점수는 높게 나올 수 있다. 예를 들어, 단순히 제곱 오차(square error)를 계량 함수로 사용할 경우 고동 형태의 출력값에 대해 모든 예측을 0으로 하더라도 높은 점수가 나온다.
- [0041] 생성단계(S20)는 의사 결정 나무 분류기를 학습하는 단계(S21)와, 모델 정확도가 기준값 이상인지 판단하는 단계(S22)를 포함한다.
- [0042] 이와 같은 생성단계(S20)를 수행한 후, 추출단계(S30)에서는 의사 결정 나무 모델을 학습시켜 원하는 만큼의 정확도를 얻은 후 모델의 진로 결정(decision path)을 가시화하여 분석대상인 제어기의 로직을 해석한다.
- [0043] 위의 과정으로 구축된 로직은 학습에 사용된 입력 패턴에 대해서만 타당성이 보장된다. 특히 과거 시점의 데이터를 입력값으로 사용하는 경우 가능한 패턴의 경우의 수가 훨씬 증가한다. 모든 패턴에 대해 출력 데이터를 생성하고 모델을 학습하는 것은 현실적으로 불가능하지만, 최대한 입력값의 다양한 시계열 패턴을 학습시키도록

한다.

- [0044] 또한, 출력 데이터의 형태가 이산값일 경우에는 생성단계(S20)에서는 알고리즘 선택 시 신경망 모델(Neural Network Model)을 부차적으로 선택할 수 있다. 본 발명에서는 신경망 모델의 학습 방법에서 활성화 함수(Activation function)로서 하드 시그모이드 함수(Hard Sigmoid Function)을 사용할 수 있으나, 반드시 이 함수에 한정되는 것은 아니다.
- [0045] 생성단계(S20)에서는 시간 간격을 조절해 가면서 신경망 모델을 학습시킨다. 예측에 대한 평가지표로 ROC AUC 점수를 사용한다. 신경망 모델은 상황에 따라 다양하게 구성할 수 있다. 본 발명에서는 크게 두 가지 모델을 사용하였다. 먼저, 현재 시점의 입력 데이터만으로 충분히 예측 가능한 경우에는 완전 연결 계층(fully connected layer)으로 이루어진 신경망 모델을 사용할 수 있다. 그 다음, 과거 시점의 데이터까지 필요한 경우에는 첫 번째 계층(layer)이 국소적 연결 계층(locally connected layer)으로 구성된 신경망 모델을 선택하여 사용할 수 있다.
- [0046] 이 경우에는 추출단계(S30)에서 학습된 신경망 모델의 가중치(weight)를 분석대상인 제어기 로직으로 번역이 쉽게 되도록, 정확도가 유지되는 범위에서 단순화한다. 그 다음, 얻은 가중치를 분석하여 래더 로직 다이어그램을 그린다.
- [0047] 다이어그램을 그리는 것이 너무 어렵다면 수식 형태로 로직을 나타낸다.
- [0048] 도 10에 도시한 출력변수 'FORWARD'에 대한 분석 결과로서 의사결정 나무 모델 학습을 통해 얻은 진로 결정 과정은 도 13의 순서도와 같이 나타낼 수 있다.
- [0049] 나무 모델을 이용시 추출단계(S30)의 예로 도 13의 결과를 래더 로직 다이어그램으로 나타내면, 도 14과 같다.
- [0050] 도 13의 진로 결정 과정은 하향식(top-down) 방식으로 따라가며 해석한다. 각 노드에서 이루어지는 이진분류(binary classification)는 해당 노드에 표시되어 있는 로직 단위의 on-off를 의미한다. 예를 들어, 도 13에서 최상단 노드에서 왼쪽 자녀 노드를 따라가며 해석하면, FORWARD(t-1)이 on이고 P1(t)이 off일 경우 현재 상태의 FORWARD(t)가 on이 된다. 한편 오른쪽 자녀 노드를 따라가면, FORWARD(t-1)이 off이고 P4(t)이 on일 경우 FORWARD(t)가 on이 된다.
- [0051] 이와 같은 분석에 따르면 'OR' 회로 아래에 FORWARD(t-1) | / |이 포함되어야 한다. 이후 검증단계(S40)에서 이 로직 단위를 제외해도 ROC AUC 점수가 그대로 유지되는 것을 확인할 수 있었고, 최종 결과에서는 이를 제외시켰다. 이 과정을 통해 실제 PLC 장치를 좀 더 간단하고 직관적인 로직으로 수정할 수 있다.
- [0052] 이처럼 출력변수 'ACC', 'DEC', 'RUN', 'STD', 'STOP'에 대한 분석을 통해 래더 로직 다이어그램을 작성한 결과를 도 15 내지 도 19에 각각 도시하였다.
- [0053] 신경망 모델에서도 부적절한 계량 함수(metric)에 대한 문제, 과적합 문제 등에 주의한다. 본 발명에서의 실험 데이터의 경우 하나의 계층으로 충분했지만, 실제 현장에서 사용되는 복잡한 로직의 경우 여러 계층이 필요할 수 있다.
- [0054] 본 발명의 일실시예에서는 의사 결정 나무 모델을 우선 실시하여 그 결과를 토대로 계층을 얼마나 쌓을지, 가중치 선택 및 전정 과정은 어떻게 하면 좋은 결과가 나오는지 등 모델 후보군을 선별한 뒤 학습 과정을 거쳤다. 이처럼 나무 결정 모델을 선행 모델로 적절히 활용하면 신경망 모델 구축 과정에서 생기는 시행착오를 줄일 수 있다.
- [0055] 과거 시점 데이터를 함께 학습하는 경우에 사용된 국소적 연결 계층의 가중치는 대부분 전정 과정을 통해 삭제할 수 있다고 가정하였다. 실제 PLC가 작동하는 방식을 보면 일부를 제외한 대부분의 과거 시점의 데이터들이 현재 시점의 결과값에 영향을 주지 않기 때문에 이런 형태의 모델링이 가능하다. 또한, 이를 통해 로직을 좀 더 이해하기 쉬운 수준으로 만들어 줄 수 있다. 전정 과정은 학습 과정에서 특정 세대(epoch) 구간에서 특정 간격으로 비중이 가장 작은 가중치들을 하나씩 제거하는 방식으로 진행될 수 있다.
- [0056] 다음은 신경망 모델 이용 시 추출단계(S30)의 일 예이다. 출력 변수 'ACC'의 경우 과거 시점의 데이터 필요없이 로직 예측이 가능하다. 이 경우 도 20과 같은 간단한 신경망 모델로 분석이 충분하다.
- [0057] 한편, 출력 변수 'STD'의 경우 과거 시점의 데이터를 함께 학습시켜야 충분한 정확도를 얻을 수 있다. 이 경우 가중치 분석을 쉽게 만들기 위해 도 21과 같은 국소적 연결 계층을 이용하며, 위에서 설명한 전정 과정을 모델 학습에 포함시킨다. 각 모델을 학습하여 얻은 가중치를 나타내면 아래의 수학적식1과 같다.

수학식 1

$$ACC(t) = \sigma(5P_1(t) + 5P_4(t) - 2.5),$$

$$DEC(t) = \sigma(5h_1(t) - 5h_2(t)).$$

[0058]

[0059] 위 식에서 h_1 과 h_2 를 정의하면 아래의 수학식 2와 같다.

수학식 2

$$h_1(t) = \sigma(-5P_4(t - 62) - 5P_1(t - 60) + 2.5),$$

$$h_2(t) = \sigma(-5P_2(t) - 5P_3(t) + 2.5),$$

[0060]

[0061] 위에서 활성화 함수 σ 는 하드 시그모이드 함수로 아래의 수학식 3으로 정의될 수 있다.

수학식 3

$$\sigma(x) = \begin{cases} 0, & \text{if } x < -2/5, \\ 1, & \text{if } x > 2/5, \\ 0.2x + 0.5, & \text{if } -2/5 \leq x \leq 2/5. \end{cases}$$

[0062]

[0063] 따라서 $P_1(t)$ 와 $P_4(t)$ 중 적어도 하나가 1이면 $ACC(t)$ 은 1이 되고, 둘 다 0이면 $ACC(t)$ 의 값은 0이 된다. 즉 출력 변수 $ACC(t)$ 는 입력 변수 $P_1(t)$ 와 $P_4(t)$ 의 OR 게이트(OR gate)에 해당되며, 이를 래더 로직으로 나타낸 것도 15이다.

[0064] 유사한 방법으로 $DEC(t)$ 에 대한 가중치 분석을 하면 도 22를 얻을 수 있다. 여기서 입력 변수 P_1 과 P_4 에는 시간 지연 모듈이 포함되어 있음을 유의한다. 한편 도 16은 나무 모델을 이용하여 얻은 출력 변수 'DEC'에 대한 래더 로직이다. 즉 같은 변수에 대한 분석이라도 다른 예측 모델을 적용하면 다른 래더 로직이 추출될 수 있다. 이는 각각의 예측이 틀렸다는 것이 아니라, 주어진 입력 변수의 패턴에 대하여 두 모델이 서로 동치인 결과를 예측한다는 것을 의미한다.

[0065] 같은 맥락으로 본 발명에서 추출된 다른 래더 로직 다이어그램 역시 실제 지능 제어장치 내부의 구조와 정확히 일치하지 않을 수는 있으나, 예측되는 출력 변수의 값에는 전혀 오차가 발생하지 않는다.

[0066] 생성단계(S20)에서 출력 데이터의 형태가 연속값일 경우의 알고리즘은 각종 기초 함수를 이용한 선형 회귀(Linear Regression) 또는 의사 결정 나무 분류기(decision tree classifier) 중 하나를 선택한다.

[0067] 다항 함수를 이용한 선형 회귀(polynomial fitting) 모델을 선택한 경우, 차수를 올려가면서 선형 회귀 모델을 적용한다. 원하는 정확도만큼 차수를 올린다. 의사 결정 나무 분류기(decision tree classifier)를 선택한 경우, 결과값의 변화량인 $y(t) - y(t-1)$ 의 값이 범주형(categorical)인 경우 혹은 적절히 반올림하여 범주형으로 만들 수 있는 경우에는 회귀 문제를 분류 문제로 바꿀 수 있다.

[0068] 결과를 해석하는 방안으로 선형 회귀 알고리즘을 적용한 경우 도출된 수식을 확인하고, 의사 결정 나무 분류기를 적용한 경우 출력 데이터를 이산값으로 간주하고 첫 번째 경우와 동일한 과정을 거친다.

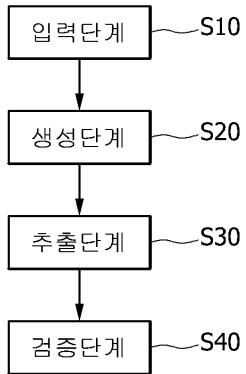
[0069] 이처럼 추출단계(S30)를 통해 PLC의 내부 로직을 확인할 수 있으며, 검증단계(S40)에서는 구축된 로직에 다시 입력값을 입력하고, 출력을 비교하여 로직이 맞는지 검증해 본다. 필요하다면 사용자의 판단 하에 불필요한 로직 단위를 제외시키고 검증을 하여 더 정확한 로직을 얻을 수 있다.

[0070] 본 발명은 상기 실시예에 한정되지 않고 본 발명의 기술적 요지를 벗어나지 아니하는 범위 내에서 다양하게 수

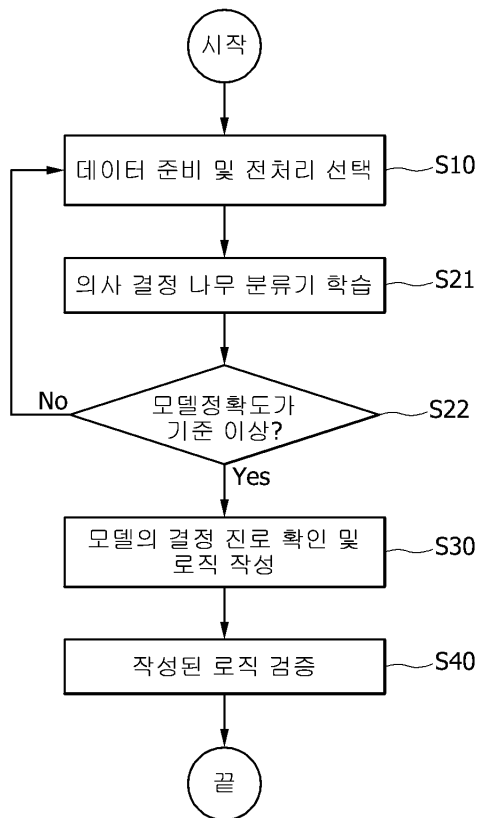
정, 변형되어 실시될 수 있음은 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 있어서 자명한 것이다.

도면

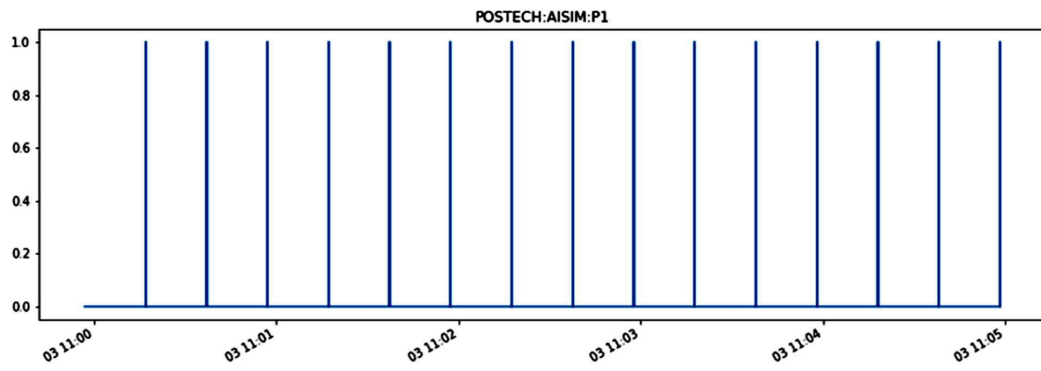
도면1



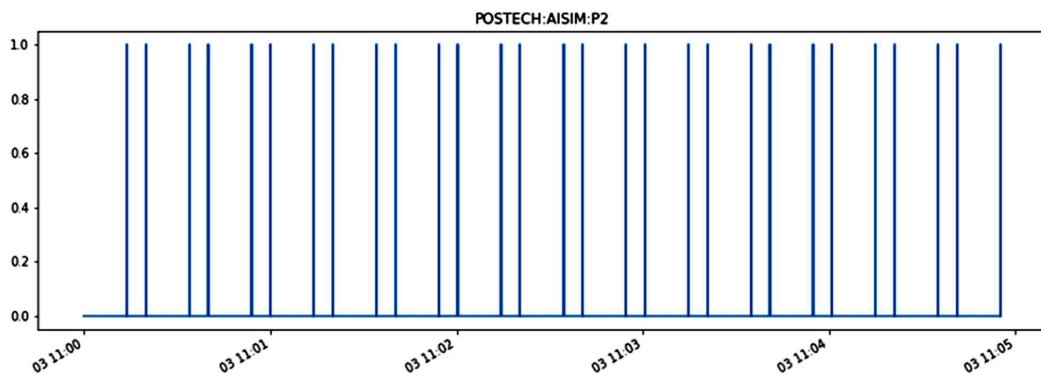
도면2



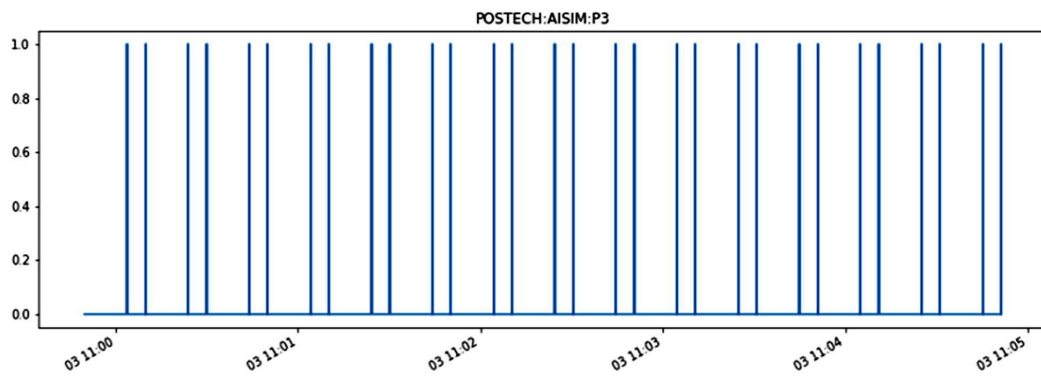
도면3



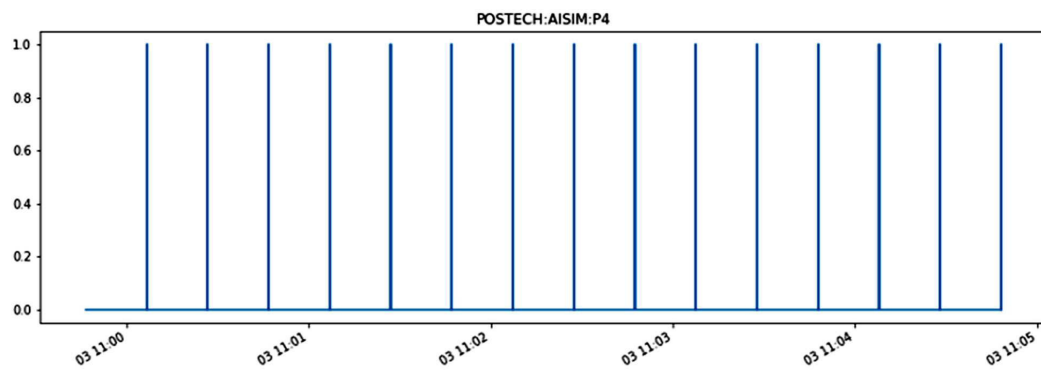
도면4



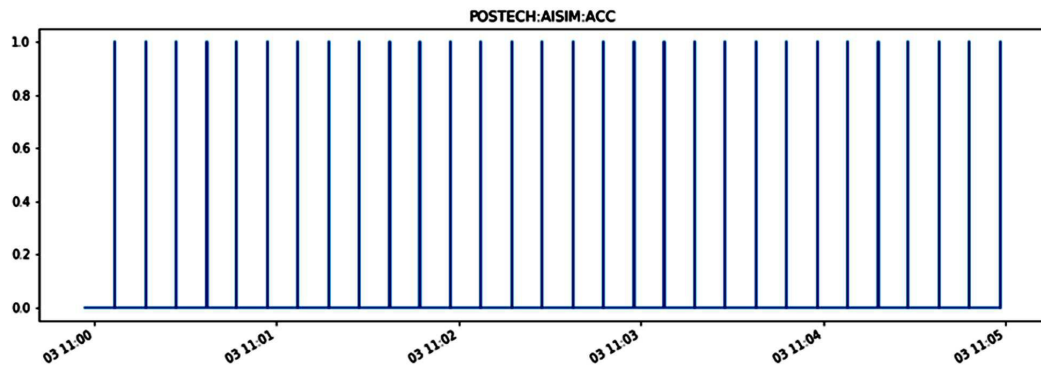
도면5



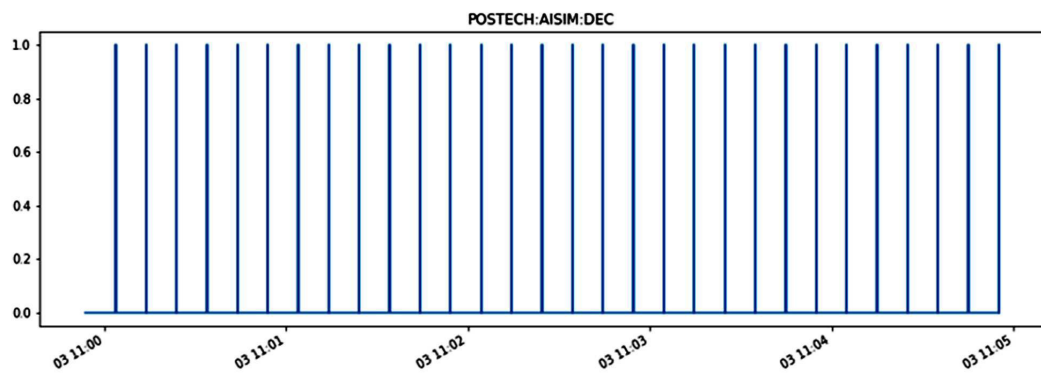
도면6



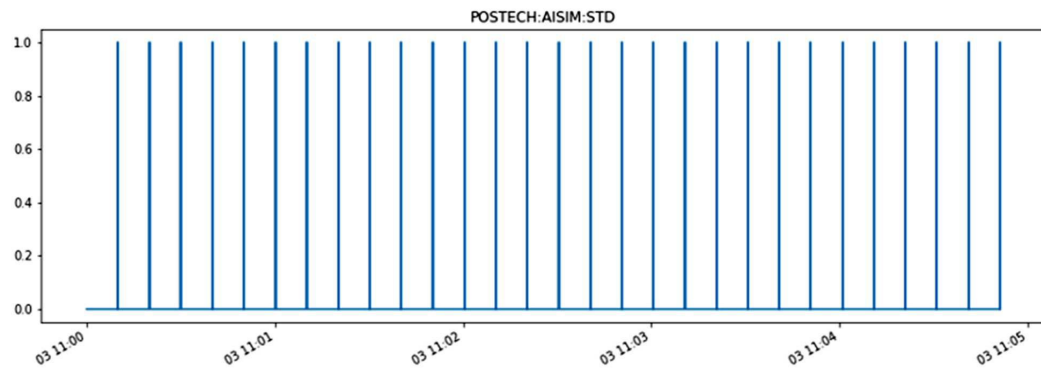
도면7



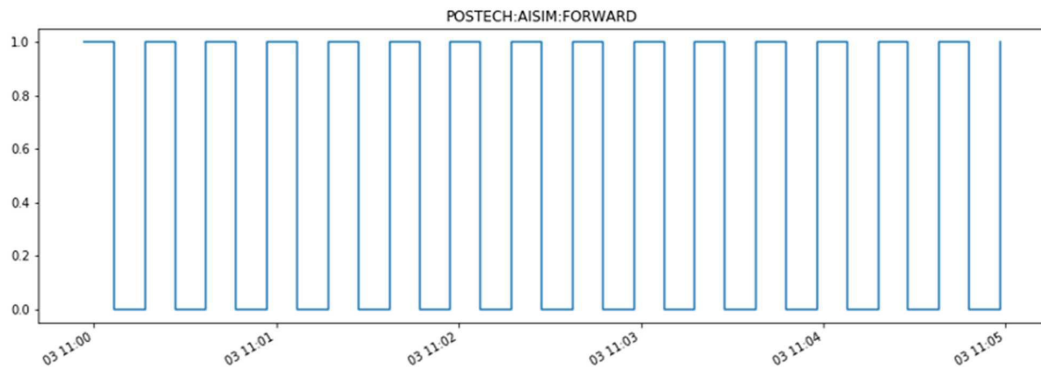
도면8



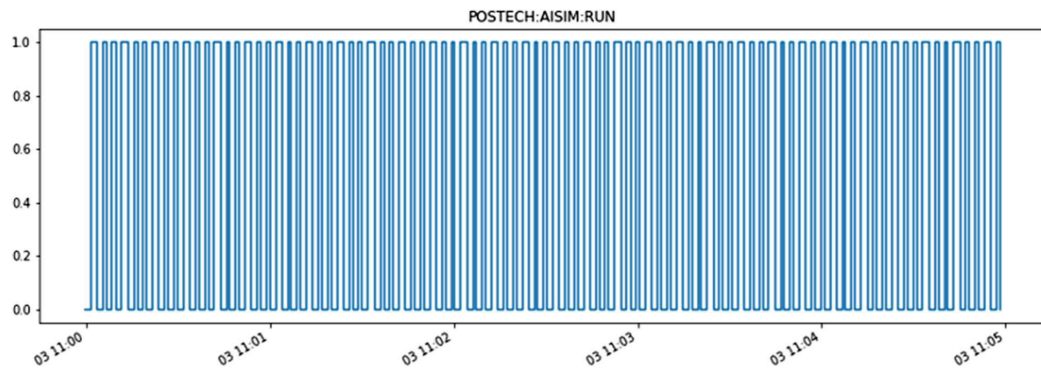
도면9



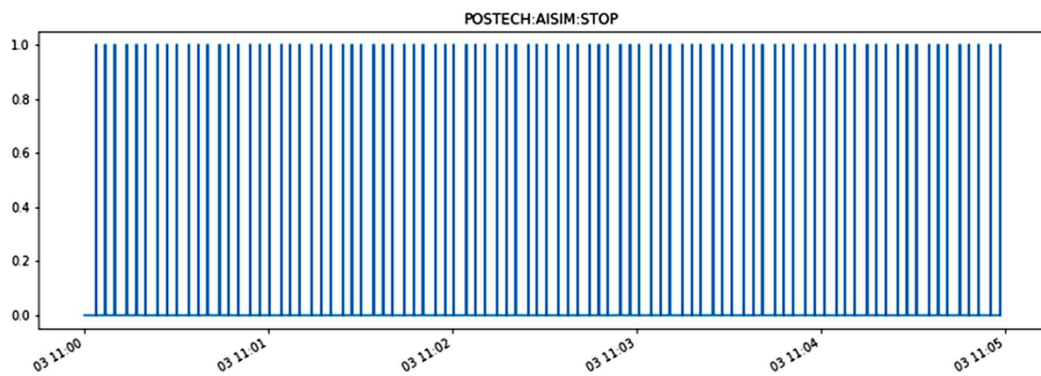
도면10



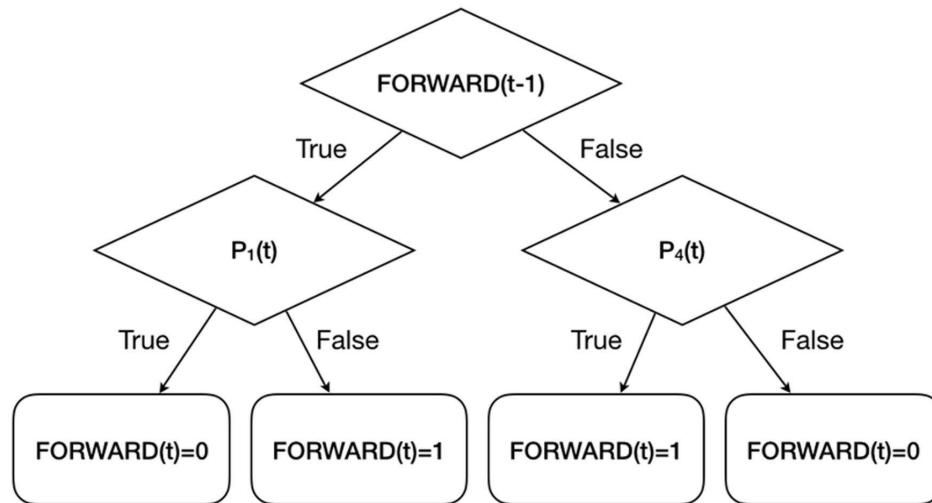
도면11



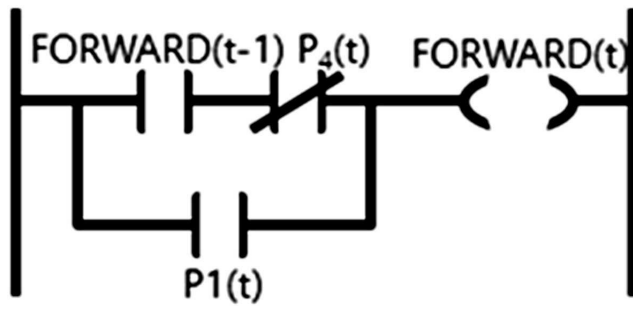
도면12



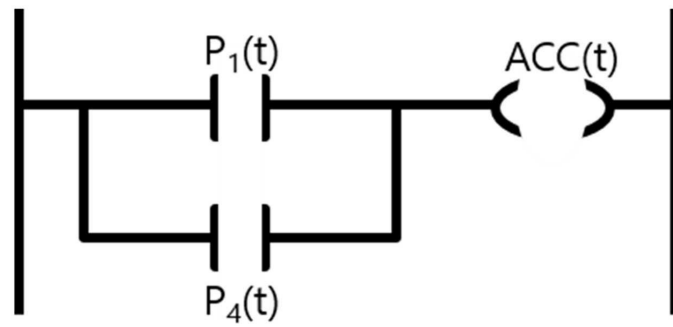
도면13



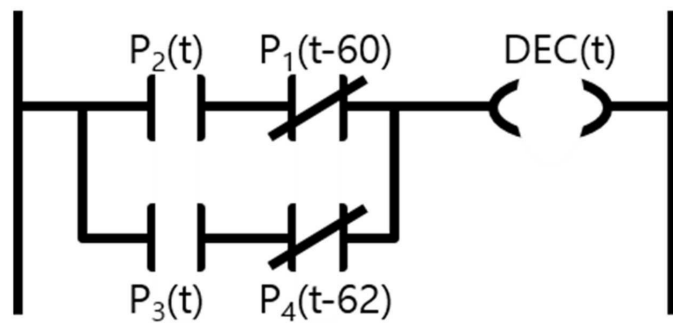
도면14



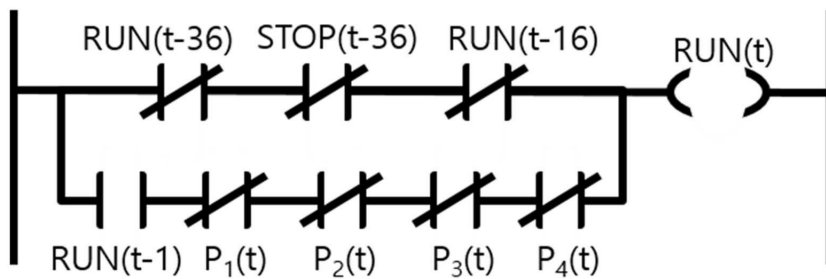
도면15



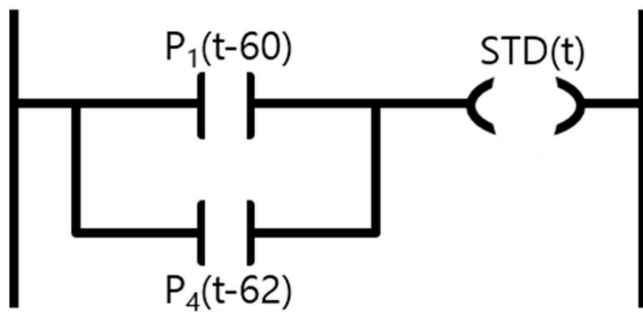
도면16



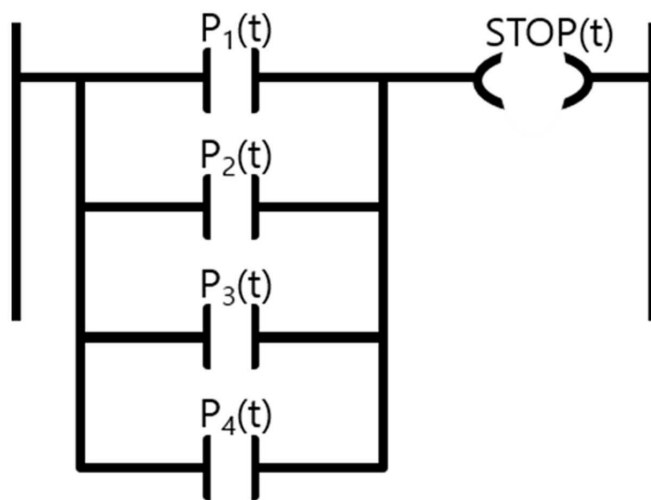
도면17



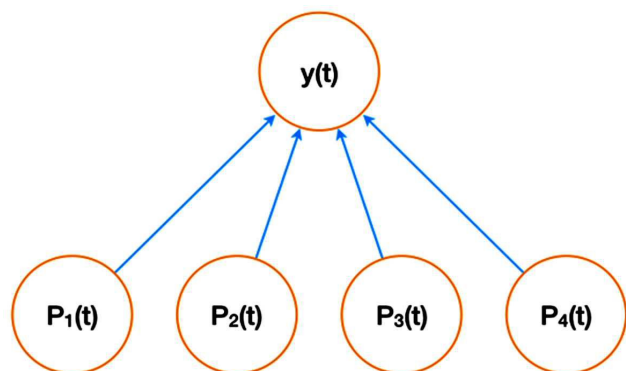
도면18



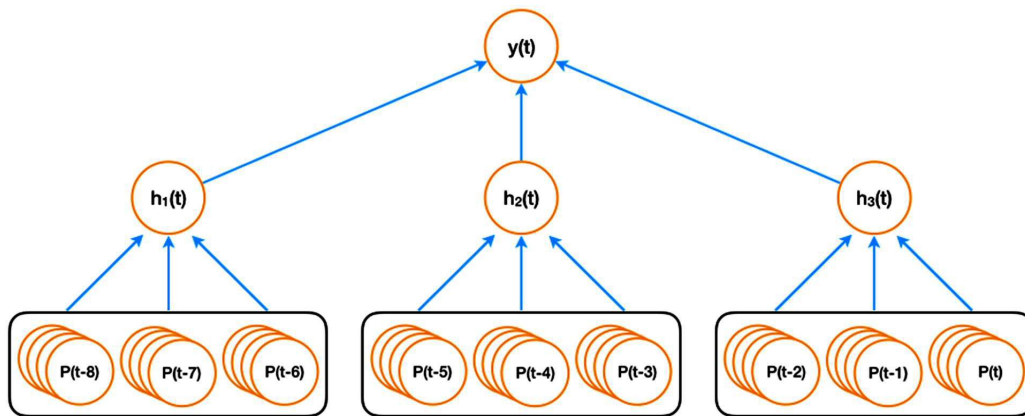
도면19



도면20



도면21



도면22

