

논문 2025-1-12 <http://dx.doi.org/10.29056/jsav.2025.03.12>

# LSTM-DQN과 Modbus TCP를 활용한 분산형 AI 기반 실내 온도제어 시스템

김홍균\*, 조현목\*\*, 이규대\*†

## Distributed AI-based indoor temperature control system using LSTM-DQN and Modbus TCP

Hong-Kyun Kim\*, Hyun-Mook Cho\*\*, Kyu-Tae Lee\*†

### 요 약

본 논문은 분산형 HVAC 시스템을 위한 중앙집중식 AI 기반 실내 온도 제어 시스템을 제안하고, AI 알고리즘과 Modbus TCP 프로토콜을 이용한 효율적 제어 방법을 실험을 통해 검증하였다. 시스템의 전체 구조는 센서, 히터, 에어컨을 제어하는 Sub-part와 AI 커멘드를 생성하는 메인파트로 구성된다. 여기에 적용된 LSTM과 DQN 알고리즘은 NVIDIA Jetson Nano Orin에 구현되었으며, 한국 천안시 지역의 2023년 온도 데이터 8,761건을 바탕으로 LSTM 모델을 개발하였다. 또한, 히스테리시스 제어 방식을 도입하여 히터와 에어컨의 빈번한 기동 문제를 완화하고 전력 효율을 향상시키는 연구를 진행하였다. Sub-part는 라즈베리파이 4B를 이용하여 온도, 습도, 전력 사용량 데이터를 InfluxDB에 실시간 저장함으로써 원격 모니터링을 지원하며, 메인파트는 웹 서버를 통해 전체 시스템의 상태와 제어 과정을 원격으로 관리할 수 있도록 하였다. 제시한 방법을 통해서 에너지 소비를 최소화하면서도 각 공간의 환경을 개별적으로 최적화하여 에너지 효율과 실내 쾌적성을 동시에 달성할 수 있다.

### Abstract

This paper proposes a centralized AI-based indoor temperature control system for distributed HVAC systems, presenting an efficient control method utilizing AI algorithms and the Modbus TCP protocol. The overall system structure consists of a Sub-part that manages sensors, heaters, and air conditioners, and a 메인파트 that generates AI-based control commands. The AI control model employing LSTM and DQN algorithms is implemented on an NVIDIA Jetson Orin Nano, trained using 8,761 annual temperature data for 2023 in the Cheonan City, Korea. Additionally, a hysteresis control method is introduced to reduce frequent switching of heaters and air conditioners, enhancing overall power efficiency. The Sub-part uses Raspberry Pi 4B to store temperature, humidity, and power usage data in real-time within an InfluxDB database, enabling remote monitoring. Meanwhile, the 메인파트 facilitates remote management of the overall system status and control process through a web server. Through the proposed approach, both energy efficiency and indoor comfort can be achieved by individually optimizing environmental conditions in each space while minimizing total energy consumption.

**한글키워드** : 인공지능, LSTM, DQN, Modbus TCP, HVAC, 스마트 빌딩

**keywords** : AI, LSTM, DQN, Modbus TCP, HVAC, Smart Building

\* 공주대학교 스마트정보기술공학과

접수일자: 2025.02.25. 심사완료: 2025.03.12.

\*\* 공주대학교 전기전자제어공학부

게재확정: 2025.03.20.

† 교신저자: 이규대(email: ktleee@kongju.ac.kr)

## 1. 서론

본 논문은 스마트 빌딩 및 소규모 건물의 온도를 AI 알고리즘에 의해서 자동 조절하는 것을 목표로 하며 쾌적한 온도 유지와 동시에 에너지를 절약하기 위한 목적이 있다. 전체 내용은 다섯 파트로 구성된다. 첫 번째 부분에서는 전체 시스템 구성을 소개하고, Modbus TCP(Transmission Control Protocol) 프로토콜을 이용한 메인파트(Main-part)와 서브파트(Sub-part)의 아키텍처 및 상호 연결 구조를 설명한다[1]. 두 번째 부분에서는 Modbus TCP 프로토콜의 작동 원리와 산업용 이더넷 네트워크에서의 역할을 기술한다[2,3]. 세 번째 부분에서는 연구에 사용된 LSTM(Long Short-Term Memory) 알고리즘과 DQN(Deep Q-Network) 알고리즘의 이론적 배경과 이들이 동적 온도 제어에 적합한 이유를 살펴본다. 네 번째 부분에서는 온도, 습도, 전력 사용량과 같은 센서 데이터를 수집하고 이를 활용해 AI 모델을 최적화하는 과정을 상세히 설명한다. 다섯 번째 부분에서는 소형 창고용 챔버를 이용한 환경 조건에서 실험을 진행하여 에너지 효율성, 온도 안정성, 사용자 편의성 측면의 성능 분석과 연구의 주요 결과를 정리하고, 향후 연구 방향과 함께 본 논문에서 제안한 시스템 구조가 향후 한국의 소형 건물 및 주거 환경에 실용적으로 적용될 수 있음을 보여준다.

## 2. 시스템 구성

### 2.1 시스템 아키텍처 및 하드웨어

그림 1은 본 논문에서 제안한 시스템의 전체 구성을 나타낸다. 시스템은 Modbus TCP/IP 프로토콜을 활용해 메인파트와 서브파트 나뉘어 있으며, 두 부분은 원격으로 연결되어 실시간 데이

터를 주고 받는다[4]. 메인파트는 NVIDIA Jetson Nano Orin 8G를 기반으로 AI 알고리즘을 실행하여 전체 시스템을 중앙에서 관리하며, 서브파트는 Raspberry Pi 4B를 기반으로 온도·습도 센서 데이터를 수집하고 히터와 에어컨을 파워릴레이(Power Relay)를 이용하여 자동 제어한다.

서브파트는 전력 제어단의 빈번한 작동을 방지하여 장치의 수명과 효율성을 높이는 히스테리시스 제어 메커니즘이 적용되었다. Jetson Nano Orin은 NVIDIA Ampere 기반 GPU와 6코어 ARM Cortex-A78AE CPU와 8GB LPDDR5 메모리, 500GB SSD를 장착한 임베디드 AI 플랫폼으로, 센서 및 제어 장치와의 연결을 위해서 USB, GPIO, I2C, SPI 등 I/O 인터페이스를 제공한다. GPIO를 이용하여 에어컨과 히터를 제어하며, I2C로 온도 및 습도 센서 데이터를 처리하고, RS485-UART 인터페이스를 이용하여 전압 및 전류를 모니터링하고 실시간으로 전력을 계산하도록 하였다.

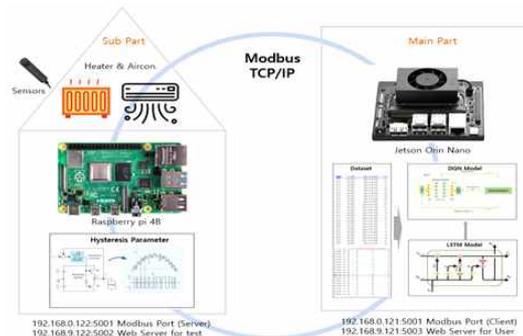


그림 1. 전체시스템 구성도

Figure 1. Overall System Architecture Diagram.

### 2.2 소프트웨어 및 데이터 처리

데이터 처리와 시스템 제어 알고리즘은 Python 3.8 및 C++로 개발되었으며, AI 모델 구현에는 TensorFlow 2.15.0 기반의 딥 러닝 프레임워크가 사용되었다[5]. 백엔드 서비스는 Flask

로 구축되었으며, Gunicorn이 애플리케이션 서버로 사용되었다. 실시간 데이터 모니터링과 시각화를 위해서 Grafana를 활용하여 그래픽 패널 기반의 직관적인 시스템 구성을 가능하게 하였다. 프론트엔드 환경은 HTML, CSS 및 JavaScript를 사용하여 구현하였다. 연구에서는 두 가지 AI 모델을 적용하였다. 첫 번째는 과거 센서 데이터를 기반으로 미래 온도 변화를 예측하는 모델로, 시계열 예측 성능이 우수한 LSTM을 사용한다. LSTM은 기존 RNN의 장기 의존성 문제를 극복하여 긴 시퀀스 내의 시간 변화에 따른 데이터 패턴을 효과적으로 학습한다.

두 번째는 HVAC(Heating, Ventilation, and Air Conditioning) 시스템의 최적 제어 파라미터를 찾기 위한 강화학습 모델로 DQN을 이용하였다. DQN 모델은 상태와 행동의 가치를 학습하여 에너지 소비를 최소화하는 동시에 사용자 편의성을 유지하는 최적의 제어 조건을 제공한다. 센서 데이터는 시계열 데이터베이스인 InfluxDB에 저장되며, 2023년 1월부터 12월까지 한국 기상청에서 측정된 천안시 지역의 연간 평균 온도 데이터 총 8,761개 데이터 포인트가 모델에 사용되었다. 실험에 사용된 센서의 오차 범위는 온도  $\pm 0.2^{\circ}\text{C}$ , 습도  $\pm 3\%$  RH, 전압 및 전류  $\pm 1\%$  이내이다[6,7].

### 3. 제어를 위한 Modbus TCP 구현

#### 3.1 프로토콜 구현 및 데이터 관리

Modbus 프로토콜은 산업 자동화에서 사용되는 통신 방식 중 하나로 간단한 구조와 드라이버 설계, 다양한 장치와의 연결 유연성 및 호환성 때문에 널리 사용된다. 본 연구에서는 중앙 AI 컨트롤러와 분산 온도 제어 장치 간의 실시간 데이터 교환을 위해서 TCP 통신 기반으로 Modbus를 구현하였다. 제안된 온도 제어 시스템

은 Modbus TCP를 이용해 클라이언트-서버 아키텍처를 구성하며, 메인파트 AI 컨트롤러가 슬레이브 역할을, 서브파트 컨트롤러가 마스터 역할을 수행한다. 각 방에 설치된 분산 장치인 서브파트 컨트롤러는 온도, 습도, 전력 소비 데이터를 모니터링하고 이를 Modbus TCP 슬레이브에서 마스터로 전송한다. 시스템 아키텍처는 슬레이브의 요청과 마스터의 응답으로 이루어진 개별 Modbus TCP 트랜잭션으로 구성되며, 최소 지연 시간과 최적의 신뢰성을 보장하기 위한 이더넷 기반 Modbus TCP 프레임 형식을 사용한다. 데이터 관리를 위해 특수 Modbus 레지스터가 통합되어 있으며, 입력 레지스터(0x04)는 온도 및 습도 판독값을, 홀딩 레지스터(0x03)는 HVAC 제어 매개변수를 저장한다. 구현에서는 0x03(매개변수 검색), 0x04(센서 데이터 수집), 0x06(동적 HVAC 조정)의 세 가지 기본 Modbus TCP 기능 코드를 활용한다.

AI 컨트롤러는 온도 모니터링을 위해 입력 레지스터 읽기(0x04) 요청을 보내고 16비트 응답을 수신하며, 메시지 프로토콜은 표준 Modbus TCP 프레임 구조를 따른다. 숫자 표현을 위해 4바이트 레지스터를 사용하고, 데이터 전송 및 변환 중 정밀도를 유지하기 위해 합의된 스케일링 요소 100을 적용한다. 예를 들어,  $25.3^{\circ}\text{C}$  판독값은 제어 알고리즘 적용 전 100을 곱해서 253 정수로 변환되며, 수신 후 100으로 나눠 25.3인 실수 값을 만든다.

#### 3.2 통신 프로토콜 및 동적 제어

통신 프로토콜은 표준 Modbus TCP 프레임 구조를 따르며, 다음과 같은 레지스터 항목들을 포함한다.

- 2바이트의 트랜잭션 ID(고유 요청 식별자)
- 2바이트의 정적 프로토콜 ID(항상 0x0000)

- 2바이트의 길이 필드(요청 크기)
- 1바이트의 단위 ID(슬레이브 장치 식별자)
- 1바이트의 기능 코드(작업 유형 지정)
- 2바이트의 레지스터 주소(데이터 위치)
- 2바이트의 레지스터 수(읽기 범위 지정)

예를 들어, 레지스터 0x000A에서 온도 데이터를 획득하기 위한 프로토콜은 다음과 같다.

요청 (마스터 → 슬레이브): [0x01] [0x04] [0x00] [0x0A] [0x00] [0x01] [CRC]

응답 (슬레이브 → 마스터): [0x01] [0x04] [0x02] [0x01] [0x90] [CRC]

프로토콜 설정 후, 시스템은 HVAC 작업의 동적 제어 메커니즘을 구현한다. AI 컨트롤러는 난방 및 냉방 강도를 조절하기 위해서 홀딩 레지스터(0x03)를 수정하며, 일반적으로 단일 레지스터 쓰기(0x06) 명령을 사용해 냉방 설정점을 조정한다. 예시 명령은 다음과 같다.

[0x01] [0x06] [0x00] [0x14] [0x00] [0x05] [CRC]

이와 같은 명령 아키텍처는 스마트 빌딩에서 실시간 온도 제어를 가능하게 한다. Modbus TCP를 AI 기반 HVAC 시스템에 통합하면 산업 제어 시스템 간의 원활한 상호 운용성, TCP 통신을 통한 빠른 온도 조정, 모듈식 설계로 인한 손쉬운 시스템 확장, 그리고 최소 하드웨어 요구 사항에 따른 비용 효율적인 대규모 시스템 구현 등 여러 기술적 이점을 제공한다.

#### 4. LSTM 및 DQN 모델링

##### 4.1 데이터셋을 이용한 LSTM 모델 구현

본 연구에서는 온도 실험을 위한 LSTM 모델 검증용 목적으로 데이터 전처리와 모델 구축 단계를 수행하였다. 우선, 실외 온도 데이터는 2023년 천안시 기상청 공개 날씨 데이터 포털을 참고

하였고 실내 온도 데이터는 사계절 평균 온도로 시물레이션하여 각각 8,761개의 시간별 데이터 포인트로 데이터셋을 구성하였다. 데이터는 날짜 순으로 정렬한 후, 각 타임스탬프에서 시간과 월 정보를 추출하고, 주기성을 반영하기 위해서 사인 및 코사인 함수를 적용하였다. 이후 모든 특징(실내·실외 온도, 시간과 월의 사인·코사인 값)은 MinMaxScaler로 정규화하였으며 지난 24시간 데이터를 입력으로 하여 다음 24시간 온도를 예측하는 시퀀스(seq\_length=24, forecast\_horizon=24)를 구성하였다. 전체 데이터셋 80%는 훈련용, 20%는 테스트용으로 분할하여 사용하였다. 두 개의 LSTM 레이어와 Dense 레이어로 구성된 모델은 25 에포크 동안 435개의 미니 배치로 훈련되었으며, 각 미니 배치는 약 31에서 70밀리초가 소요되어 에포크(Epoch) 당 평균 30초가 걸렸다. 훈련 손실과 검증 손실은 초기 0.003에서 점차 감소하여 약 0.0024로 수렴하였으며, 역변환 및 상한 온도 제한, 지수 평활법을 적용한 후 그래프로 모델의 안정적인 학습 결과를 확인하였다.

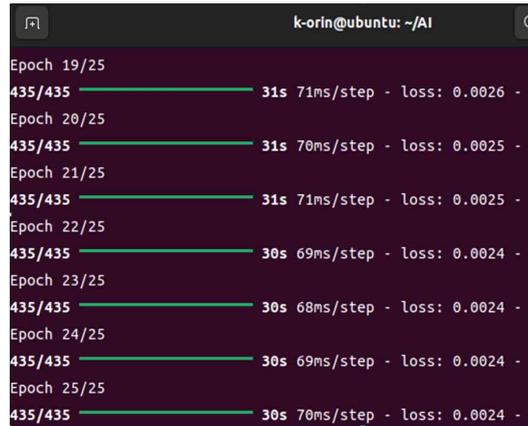


그림 2. LSTM 훈련 과정 터미널 출력 화면  
Figure 2. Terminal Output Screen for LSTM Training Process.

### 4.2 LSTM 모델 검증

그림 3의 그래프는 LSTM 모델이 예측한 24시간 실내 온도 변화와 실제 측정된 온도 변화를 나타낸다. x축은 현재 시간으로부터의 경과 시간을, y축은 온도(°C)를 나타내며, 실선은 지수 평활법으로 처리된 LSTM 예측 온도 곡선을, 점선은 테스트 세트의 실제 온도 데이터를 보여준다. x축 범위는 0에서 24는 특정 시점 이후 24시간 동안의 예측 온도와 실제 온도가 효과적으로 적용되었음을 시각화한 것이다. LSTM 모델은 이전 24시간의 실내 및 실외 온도 데이터를 기반으로 다음 시간 단계의 실내 온도를 예측한다. 이 예측 결과는 DQN 모델에 통합되어, 실시간 추가 환경 변수와 사용자가 지정한 온도 설정점을 포함하는 6차원 상태벡터 LSTM 출력값인, 평균 습도, 실내 온도, 실외 온도, 전력 소비, 오프셋 값으로 구성되었다. 사용자 오프셋과 현장 설정 온도는 JSON 파일을 통해서 수동으로 구성되어 즉시 반영되도록 하였다. DQN은 이 상태를 처리하여 Q-러닝 업데이트를 수행하고 최적 보상 값을 계산하였으며, 이를 LSTM 예측 온도와 가중 평균하여 최종 제어 온도인 LSTN-DQN 제어값

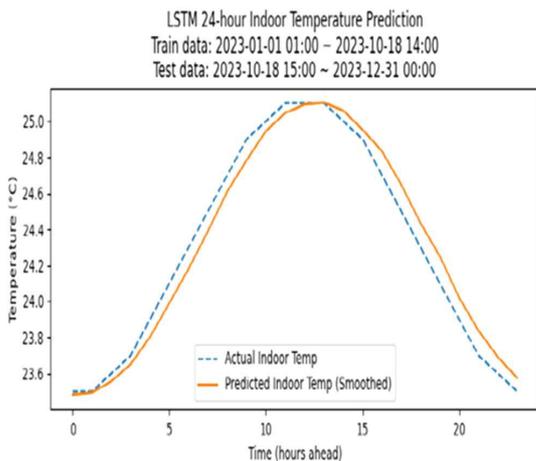


그림 3. LSTM 훈련 결과 검증 그래프  
Figure 3. LSTM Training Results Validation Graph.

을 결정하였다. 이 값은 원격으로 수신된 측정 온도를 기반으로 임계 범위 내에서 히터와 에어컨을 작동시켜 실내 온도를 조절하였다 [8,9,10,11].

### 4.3 DQN 모델 제어 검증

그림 4는 제어 섹션의 터미널 상태 창을 보여준다. 해당 창에는 실험 창고의 높이 차 (채널 간 약 70cm)에 따른 시간별 온도, 습도, 히스테리시스 설정에 따른 히터와 에어컨의 작동 상태, 그리고 Modbus TCP를 통한 원격 제어 상태가 표시된다. 또한, DQN을 통해서 생성된 최종 결과를 기반으로 AI 메인 시스템의 LSTM 출력, 오프셋, 제어 방식, MSE(Mean Squared Error) 값이 포함되어 있다. 여기에서 DQN 알고리즘은 실시간 변화 데이터에 기반한 Q-러닝 업데이트를 통해서 최적의 제어 조건을 산출하는 역할을 한다[12].

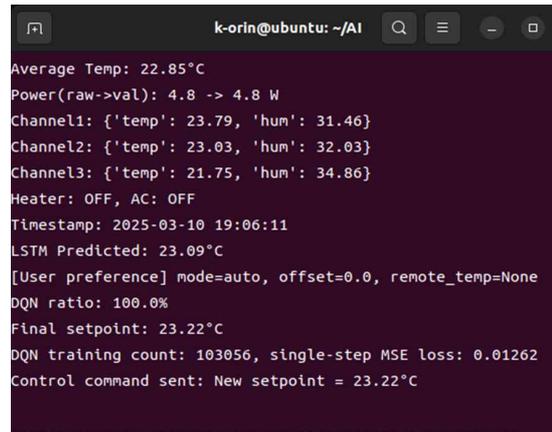


그림 4. 시스템 상태 및 DQN 최종 출력 터미널  
Figure 4. System Status and DQN Final Output Terminal.

## 5. 시스템 실험

그림 5는 본 논문의 알고리즘 개발을 위한 테스트

환경을 보여준다. 테스트는 3.3평방미터의 조립식 실험 공간인 챔버에서 수행되었으며, 이 공간에는 소형 에어컨(AC220V/2.6A, 550W), 히터(AC220V/4.5A, 1000W), 공기 순환 팬(5V/0.6A, 3W)이 설치되어 일반적인 실내 환경을 시뮬레이션하였다. 이 장치들은 NVIDIA Jetson Nano Orin 8G 모델을 사용하여 제어되었으며, 해당 모델은 Ubuntu 22.04 기반 운영 체제와 JetPack 6.1 SDK를 통해서 작동하는 메인 AI 제어 장치로 활용되었다. 이러한 설정은 다양한 HVAC 제어 작업의 실시간 처리 및 통합을 가능하게 하였다[13].

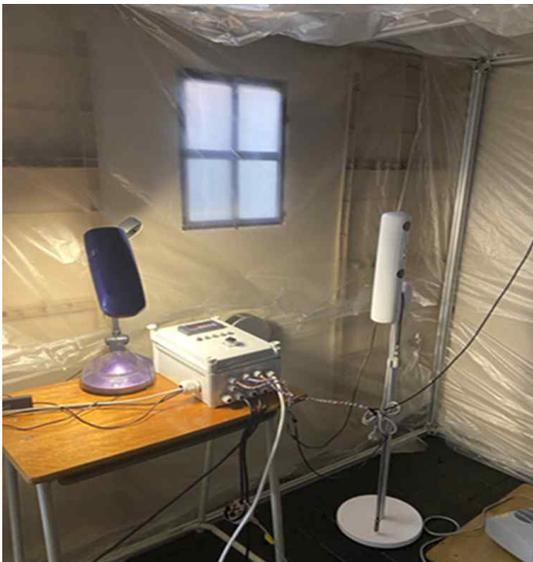


그림 5. 실험용 챔버 내부 모습  
Figure 5. View of the Experimental Chamber Interior.

온도 및 습도 센서는 공간 전체의 평균 온도를 측정하고, 전류 및 전압 센서는 실시간 전력 소비를 측정한다. 센서 데이터는 실시간으로 AI 제어 파라미터를 산출하는데 사용된다. NVIDIA Jetson Nano Orin에 탑재된 AI 알고리즘은 TensorRT를 통한 모델 최적화로 신속한 추론을

수행하여 제어 결정의 속도와 정확도를 향상시켰다. 데이터 처리 및 모델 구현에는 Python이 사용되었고, C/C++는 실시간 장치 제어에 적용되어 효율성을 보장하였다. 또한, AI 모델의 병렬 처리는 NVIDIA CUDA를 활용하여 계산 속도를 높였다[17]. 메인 파트와 서브 파트의 상태 및 제어 값을 저장하고 모니터링하기 위해서 2가지의 프로그램을 만들었다 첫째로 그림 6은 서브 파트 패널로, 온도, 습도, 전력 소비 등 시스템 상태를 실시간으로 표시하고 있으며 그림 7은 메인 파트에서 센서 데이터, 전력 소비, 온도, 습도 등 최신 센서 측정값과 AI 예측 온도 제어 값, 그리고 시스템 제어 값을 나타낸다[14].



그림 6. 서브 파트 모니터링 화면  
Figure 6. 서브파트 Monitoring Screen.



그림 7. AI 메인 파트 모니터링 화면  
Figure 7. AI 메인파트 Monitoring Screen.

## 6. 결 과

표 1은 2025년 3월 4일 00:00부터 약 1시간 간격으로 AI가 계산한 최종 설정점 온도인 LSTM-DQN과 측정된 실제 평균 실내 온도를 보여준다. 표의 데이터에 따르면 최종 설정점 온도는 시간이 지남에 따라 증가하여 오전 11시경 최고 23.51°C에 도달하였고, 실제 온도는 약 0.1°C 오차 범위 내에서 안정적인 추세를 보였다. 이는 알고리즘이 효과적으로 적용되어 DQN이 LSTM 예측값을 적용하고 사용자 설정, 전력 소비, 습도 등을 고려한 최적의 제어를 수행함을 보여준다. 향후 연구에서는 하나의 AI 메인 컨트롤러를 통해서 다수의 하위 컨트롤러로 여러 공간을 동시에 제어하는 실험을 진행할 계획이다. 또한, 실제 거주자가 느끼는 온도를 반영하여 AI 모델의 예측 정확도를 객관적으로 검증하는 동시에, 수집된 센싱 데이터를 기반으로 온도와 에너지 사용 패턴을 분석하여 보다 정교한 HVAC 제어 시스템으로 업그레이드할 방안을 모색할 예정이다.

표 1. LSTM-DQN 제어에 따른 실측 온도  
Table 1. Measured Temperature under LSTM-DQN Control.

측정 시점	LSTM-DQN제어값	측정온도
2025-03-04 00:13:42	21.62	21.65
2025-03-04 01:13:47	21.65	21.65
2025-03-04 02:13:49	21.66	21.71
2025-03-04 03:13:53	21.69	21.77
2025-03-04 04:13:57	21.77	21.82
2025-03-04 05:13:58	21.71	21.78
2025-03-04 06:14:02	21.76	21.81
2025-03-04 07:14:07	21.70	21.75
2025-03-04 08:14:10	21.80	21.85
2025-03-04 09:14:13	21.79	21.84
2025-03-04 10:14:18	21.81	21.90
2025-03-04 11:00:24	23.51	23.31

## 감사의 글

이 논문은 2020년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 연구임.(2020R111A3070744)

## 참 고 문 헌

- [1] Smith, R.G. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Trans. Comput.* 1980, 29, pp. 1104 - 1113. DOI: 10.1109/TC.1980.1675516
- [2] Modbus Organization. MODBUS Messaging on TCP/IP Implementation Guide: V1.0b. Available online: [https://www.modbus.org/docs/Modbus\\_Messaging\\_Implementation\\_Guide\\_V1\\_0b.pdf](https://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf).
- [3] Modbus Organization. Modbus Application Protocol Specification V1.1b. Available online: [https://modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b3.pdf](https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf).
- [4] Si, Y.; Korada, N.; Ayyanar, R.; Lei, Q. A high performance communication architecture for a smart micro-grid testbed using customized edge intelligent devices (eids) with spi and modbus tcp/ip communication protocols. *IEEE Open J. Power Electron.* 2021, 2, pp. 2 - 17. doi.org/10.1109/OJPEL.2021.3051327
- [5] <https://www.tensorflow.org>.
- [6] Henze, G.P.; Schoenmann, J. Evaluation of Reinforcement Learning Control for Thermal Energy Storage Systems. *HVAC R Res.* 2003, 9, pp. 259 - 275. DOI: 10.1080/10789669.2003.10391069
- [7] Homod, R.Z.; Yaseen, Z.M.; Hussein, A.K.; Almusaed, A.; Alawi, O.A.; Falah, M.W.; Abdelrazek, A.H.; Ahmed, W.; Eltaweel, M. Deep clustering of cooperative multi-agent reinforcement learning to

optimize multi chiller HVAC systems for smart buildings energy management. J. Build. Eng. 2023, Volume 65, pp. 105689. DOI: 10.1016/j.jobe.2022.105689

[8] Torres, J.; Martí'nez-Álvarez, F.; Troncoso, A. A Deep LSTM Network for the Spanish Electricity Consumption Forecasting. Neural Comput. Appl. 2022, Volume 34, pp. 10533 - 10545. doi.org/10.1007/s00521-021-06773-2

[9] Dubey, A.K.; Kumar, A.; García-Díaz, V.; Sharma, A.K.; Kanhaiya, K. Study and analysis of SARIMA and LSTM in fore-casting time series data. Sustain. Energy Technol. Assess. 2021, Volume 47, pp. 101474. DOI: 10.1016/j.seta.2021.101474

[10] Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM networks. In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Montreal, QC, Canada, 31 July - 4 August 2005; Volume 4, pp. 2047 - 2052. DOI: 10.1109/IJCNN.2005.1556215.

[11] Ismail, M.M.B. Insult detection using a partitional CNN-LSTM model. Comput. Sci. Inf. Technol. 2020, 1, pp. 84 - 92. doi.org/10.11591/csit.v1i2.p84-92

[12] Devarajan, G.G.; Nagarajan, S.M.; Ramana, T.; Vignesh, T.; Ghosh, U.; Alnumay, W. DDNSAS: Deep reinforcement learning based deep Q-learning network for smart agriculture system. Sustain. Comput. Inform. Syst. 2023, Volume 39, pp. 100890. doi.org/10.1016/j.suscom.2023.100890

[13] <https://developer.nvidia.com/sdk-manager>.

[14] <https://developer.nvidia.com/cuda-toolkit>.

저 자 소 개



김홍균(Hong-Kyun Kim)

2006.8 충북대학교 전기공학과 박사  
 2000.1-2018.8 (주)다스파워 CEO  
 2006.9-2009.12 전남대학교 겸임교수  
 2013.8-2015.5 이화여자대학교 연구교수  
 2006.9-2022.9 국립한밭대학교 겸임교수  
 2019.3-현재 K-ICT창업멘토링센터 멘토  
 2013.11.27 미래창조과학부장관 표창  
 <주관심분야> SLLM, MMAI, Edge AI



조현묵(Hyun-Mook Cho)

1995.2 고려대학교 전자공학과 박사  
 2015.8~2016.7 Georgia Tech. 방문 교수  
 1995.9~현재 공주대 전기전자제어공학부 교수  
 <주관심분야> 회로설계, 인터페이스IC



이규대(Kyu-Tae Lee)

1991 고려대 전자공학과 박사  
 1992-현재 국립공주대학교 스마트정보기술 공학과 명예교수  
 <주관심분야> 신호처리, VLC, 저작권보호, 임베디드 시스템, 상황인식 및 학습