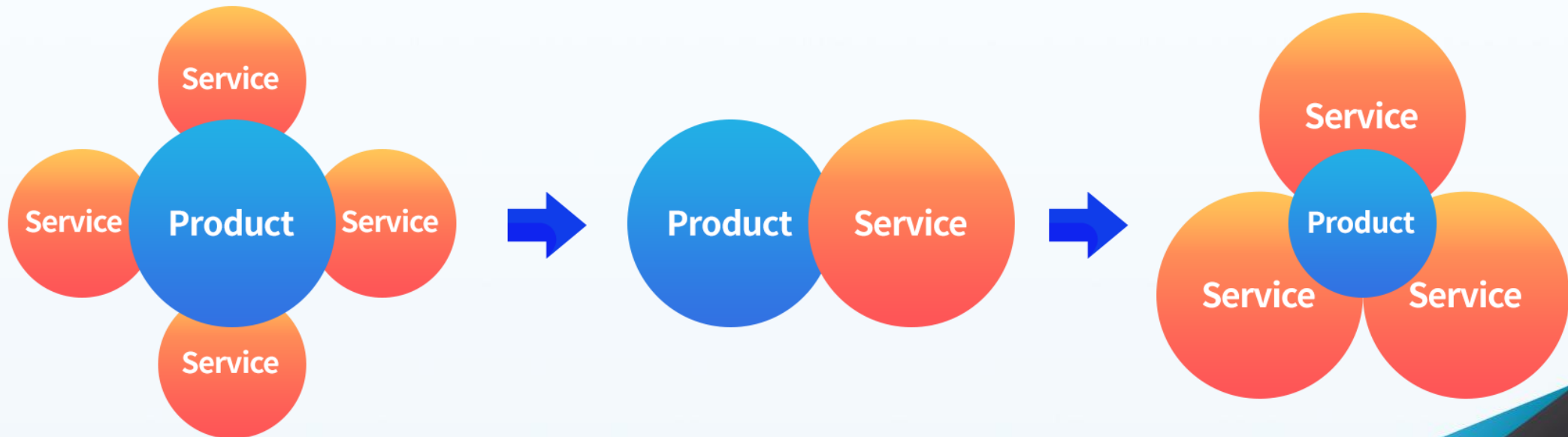


## IoT의 시대 - 서비타이제이션

- 서비타이제이션은 제품과 서비스의 결합(product servitization), 서비스의 상품화(service productization), 그리고 기존 서비스와 신규 서비스의 결합 현상을 포괄하는 개념



## IoT의 시대 – 서비타이제이션 사례



중장비 IoT



블랙박스 IoT

## 우리회사의 제품 및 서비스를 어떻게 확장할 것인가?

하드웨어를 만들고  
판매하는 회사



기존 및 신규 하드웨어 IoT를  
어떻게 구현해서 서비스 할 것인  
가?



본 영상의 2부와 3부를 보세  
요

산업별로 서비스를  
제공하는 회사

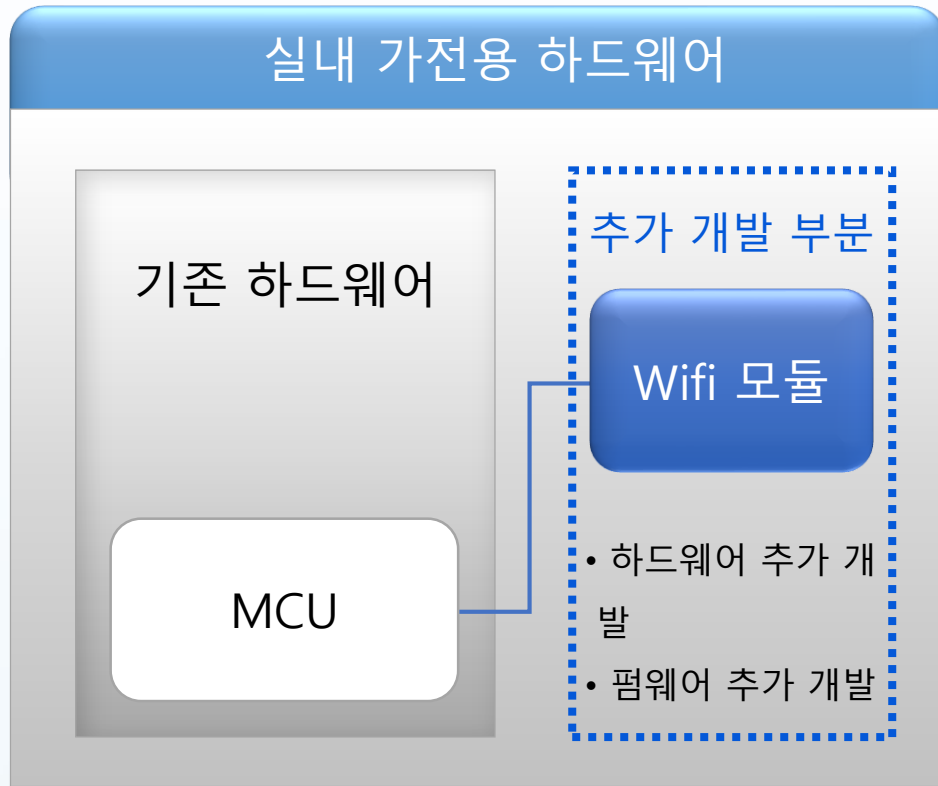


기존 센서 및 기기를 이용하여서  
내 서비스에 어떻게 적용할 것인  
가?

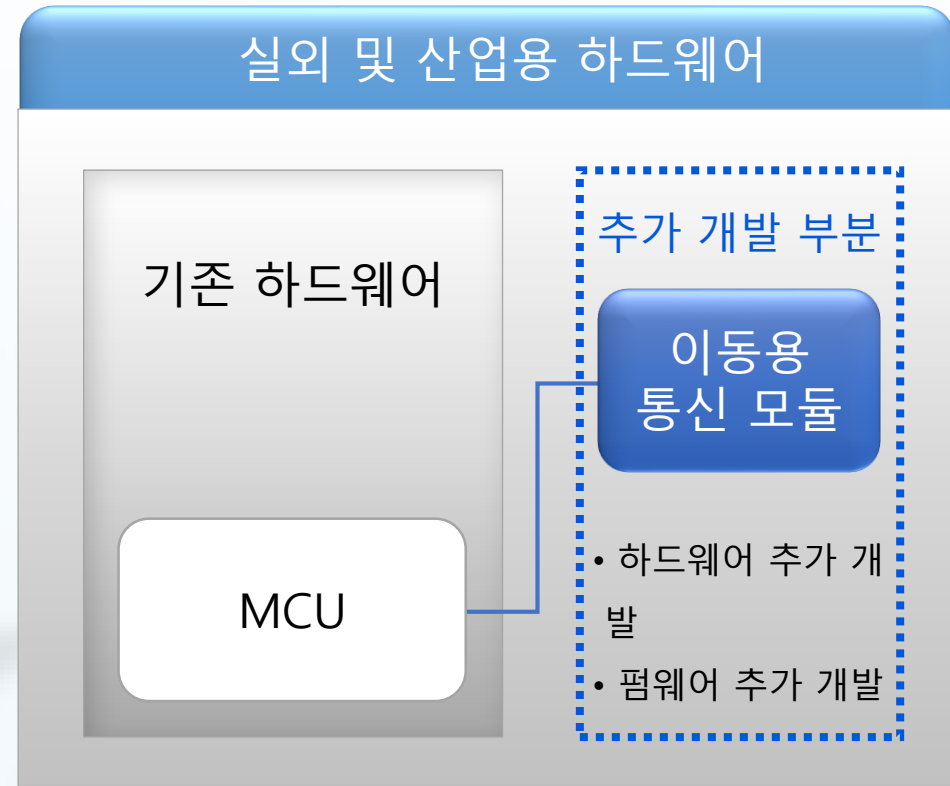


본 영상의 3부를 보세요

# 하드웨어를 만드는 제조회사의 경 우

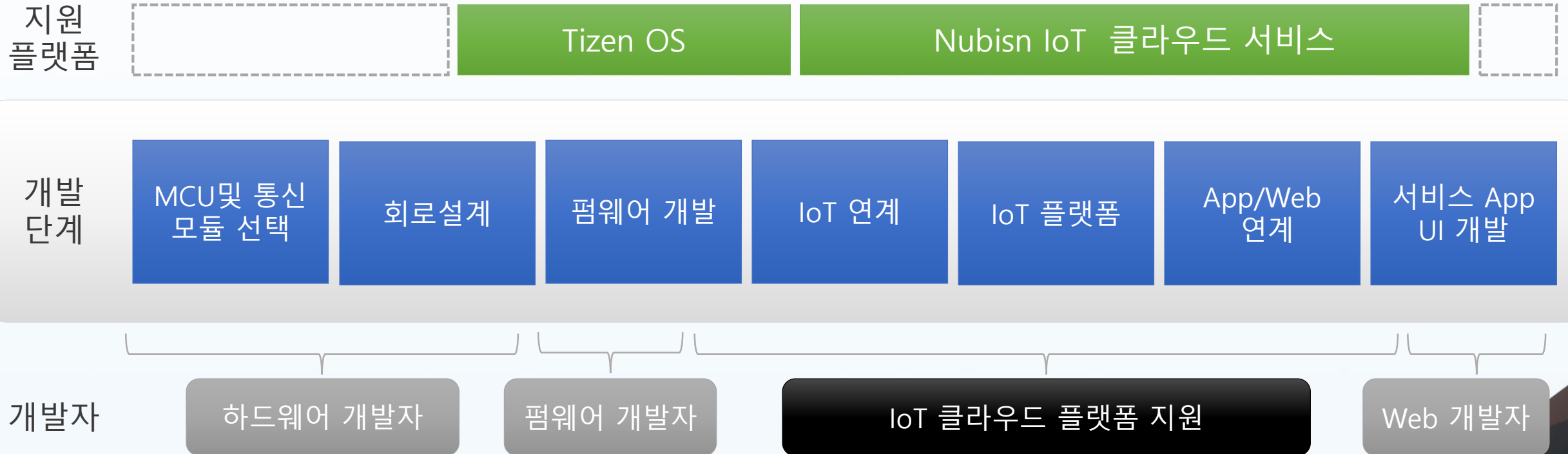


- 예)
- LED 조명
  - 이온 정수기 등



- 예)
- 태양광 전력 수집 장치
  - 빌딩 화재감지기

# 신규 기기 기반의 IoT 서비스 전체 개발 과정



# Why Nubison IoT?



# IoT의 현실적 문제



단순 온도 값을 모바일 기기 에서 보기 위해 **상당히 많은 개발 과정 필요!**

## IoT의 현실적 문제 (운영시)



개발 완료



서비스 운영



유지/보수



업데이트



IoT 서비스 운영

IoT 시스템개발 이후 부터 본격적인 문제!!





# NUBISON IoT 소개

## [IoT Service Provider]

### [Hardware Providers]

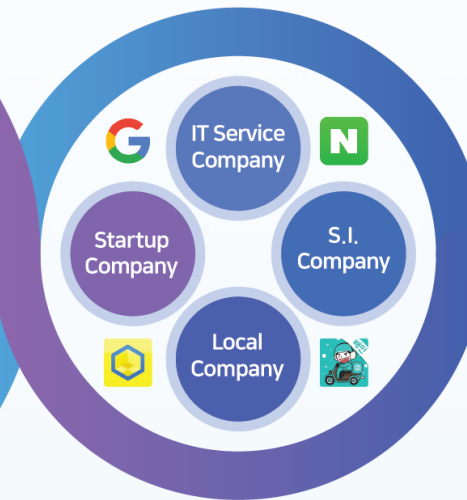


Equipment  
Sensor  
Actuator  
AI Device  
Moving Asset  
etc.



IoT Connection  
IoT Rule Processing  
IoT UI/UX Making  
IoT Datamining  
IoT Data Analysis  
etc.

### [Service Providers]



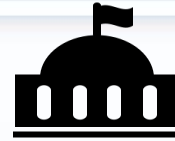
IoT based App/Web Service  
Wide area Monitoring & Control  
IoT S.I. Project  
Startup IT Service  
Industrial Expert Service  
etc.



**Various  
IoT Services**

# NUBISON IoT 소개

End User



End Service

개별 IoT 서비스

다양한 IoT 서비스

Huge Platform 서비스

AI Speaker App

Smart Home App

스마트 팩토리

스마트 시티



Hardware &  
Domain

산업용 장치 업체

중소중견 가전업체

일반 서비스 업체



## NUBISON IoT의 특징점

1. SW/HW 및 IoT에 대한 전문 지식이 없는 사람들도 쉽고 편리하게 IoT를 활용한 자신만의 서비스를 구성(Servitization) 할 수 있도록 **클라우드 웹 화면에서 Non-Coding 방식**으로 IoT 시스템을 구축 할 수 있습니다.
2. 클라우드 서비스 (PaaS & SaaS) 방식이므로 IoT시스템을 구축하는 데 있어 일반적인 개발방식 대비 **초기 구축비용이 매우 저렴하고 사용하는 만큼의 이용요금을 지불하므로 매우 경제적** 입니다.
3. 기본 SaaS 서비스 이외에도 **RestFul API**를 제공하여 원하는 서비스에 최적화 된 결과물을 만들 수 있습니다.

# 기술적 특징



초기 고객유입 가능



실사용자 중심 서비스



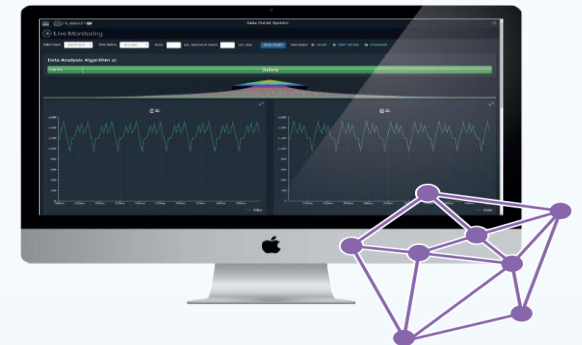
신규가치 생성



Cloud 기반 System



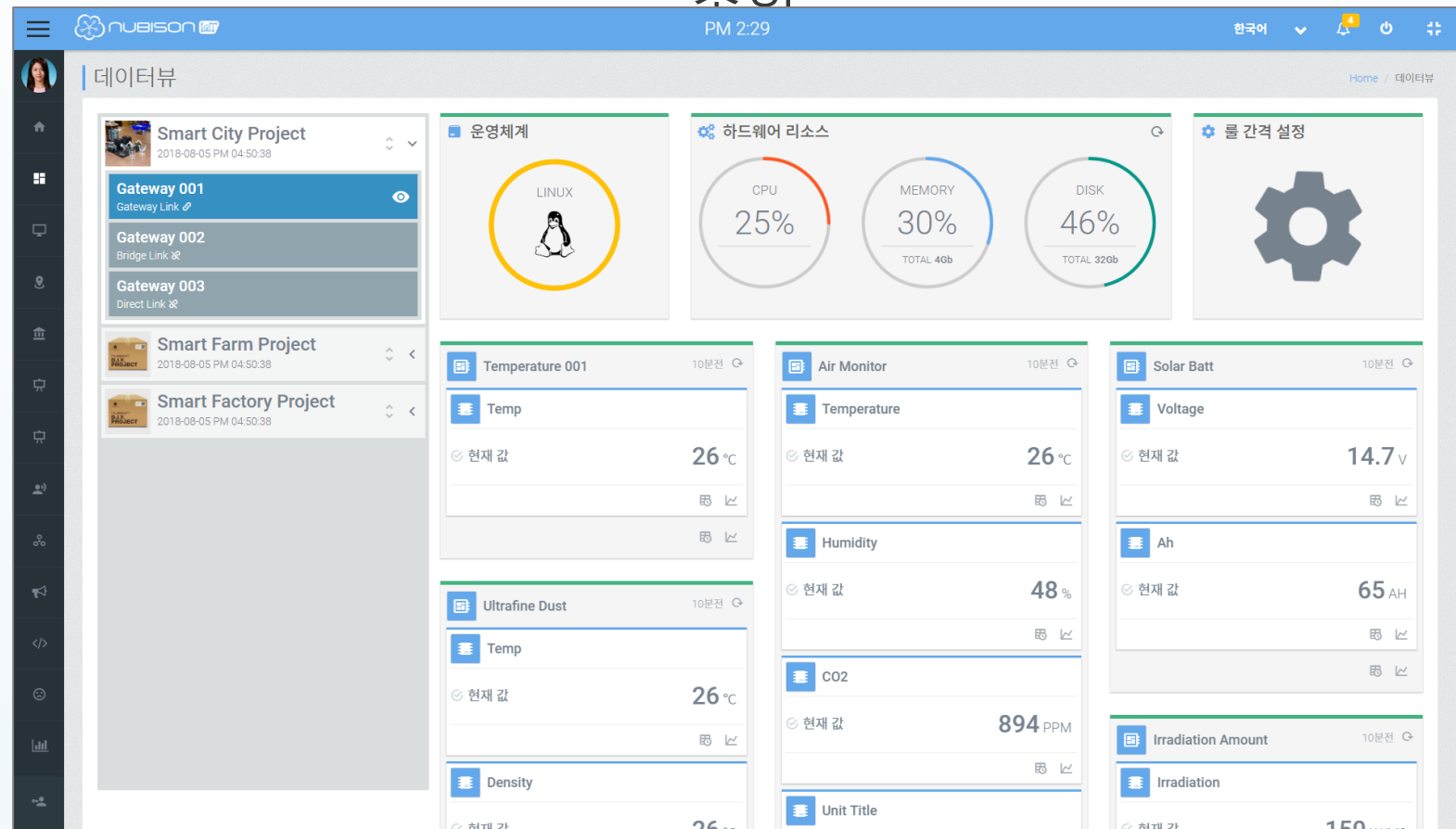
PreBuilt 서비스



IoT 특화 데이터 분석

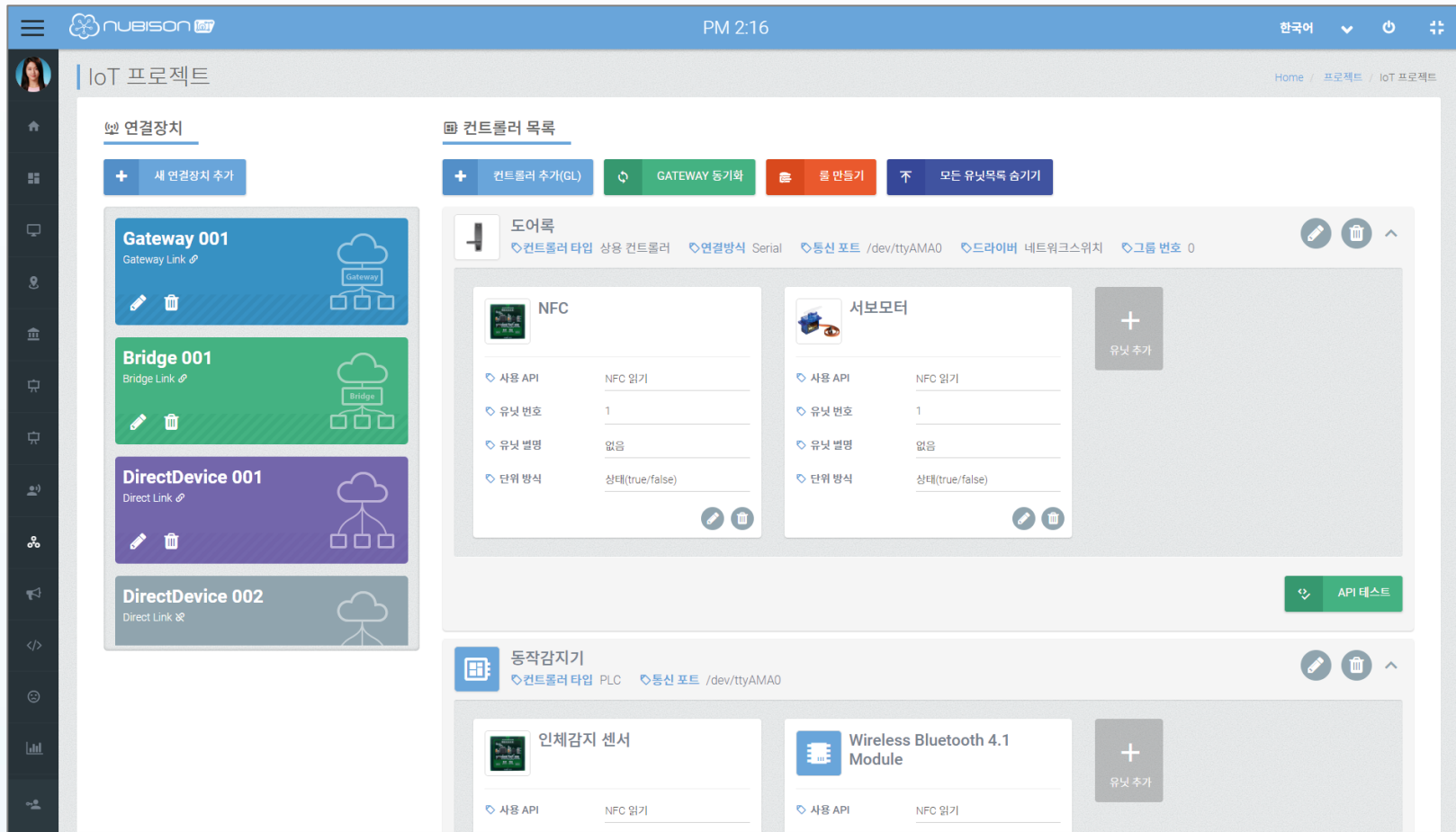
# 데이터 뷰

- 등록된 디바이스의 현재 상태 및 최신 데이터 조회 / 디바이스의 LAW 데이터 및 통계 데이터 조회



# 프로젝트

- 클라우드에 디바이스를 등록



The screenshot displays the NUBISON IoT Project Management Interface. The top navigation bar includes the NUBISON logo, the time (PM 2:16), and a language dropdown set to Korean. The main header shows the current project as 'IoT 프로젝트'.

On the left sidebar, there are icons for various functions: Home, Device, Gateway, Bridge, Direct Device, and a search icon.

The main content area is divided into two primary sections:

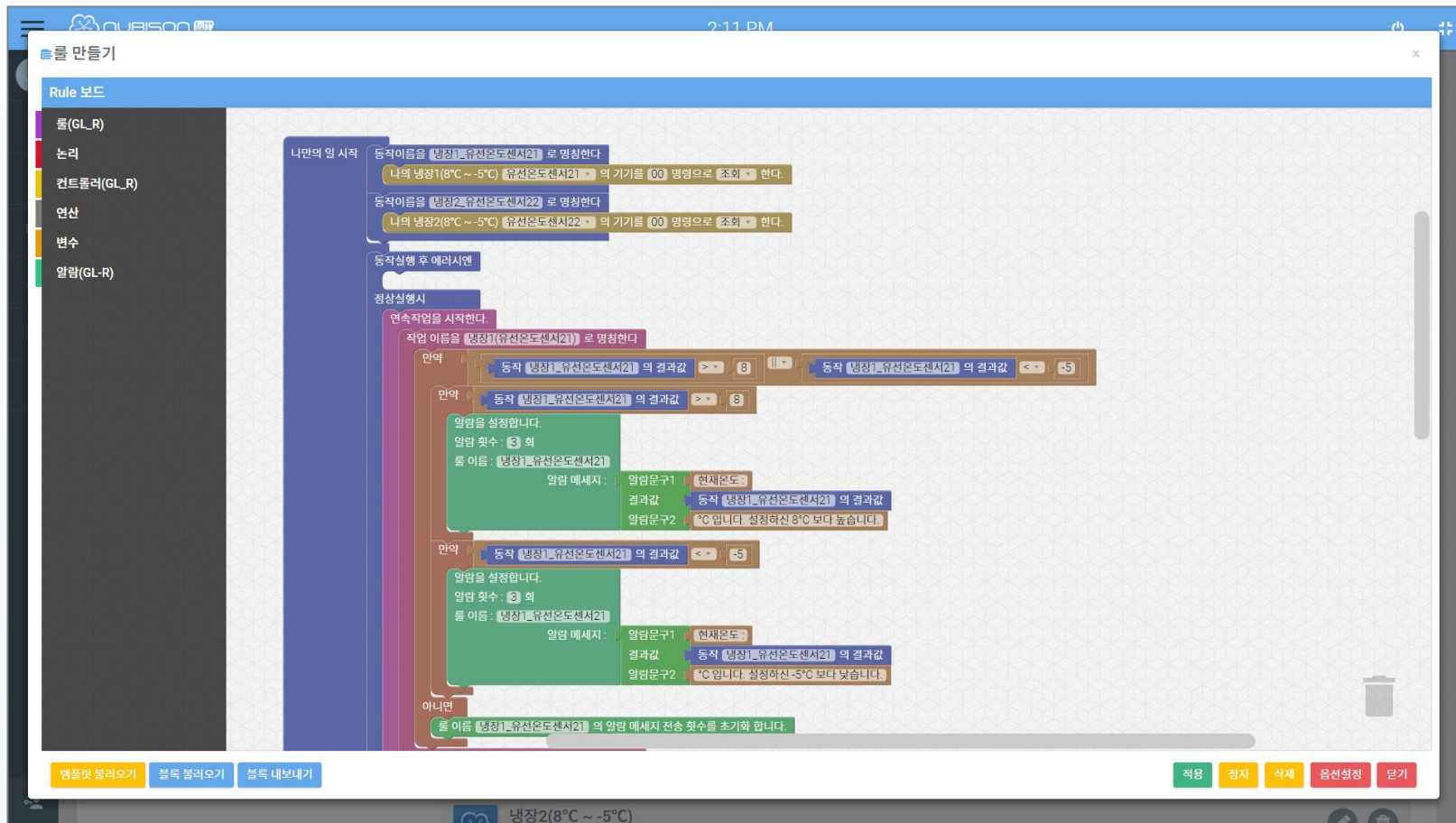
- 연결장치 (Connected Devices):** This section lists four devices:
  - Gateway 001:** Gateway Link. Includes edit and delete icons.
  - Bridge 001:** Bridge Link. Includes edit and delete icons.
  - DirectDevice 001:** Direct Link. Includes edit and delete icons.
  - DirectDevice 002:** Direct Link. Includes edit and delete icons.
- 컨트롤러 목록 (Controller List):** This section shows a list of controllers with buttons for '컨트롤러 추가(GL)', 'GATEWAY 동기화', '물 만들기', and '모든 유닛목록 숨기기'. Below this, there are two panels:
  - 도어록 (Door Lock):**
    - NFC:**
      - 사용 API: NFC 읽기
      - 유닛 번호: 1
      - 유닛 별명: 없음
      - 단위 방식: 상태(true/false)
    - 서보모터 (Servomotor):**
      - 사용 API: NFC 읽기
      - 유닛 번호: 1
      - 유닛 별명: 없음
      - 단위 방식: 상태(true/false)
  - 동작감지기 (Motion Detector):**
    - 인체감지 센서 (Human Body Detection Sensor):**
      - 사용 API: NFC 읽기
    - Wireless Bluetooth 4.1 Module:**
      - 사용 API: NFC 읽기

Each device or module panel includes a '+ 유닛 추가' (Add Unit) button and an 'API 테스트' (API Test) button.



# IoT 룰

- 디바이스의 데이터에 기반한 룰 로직 설정

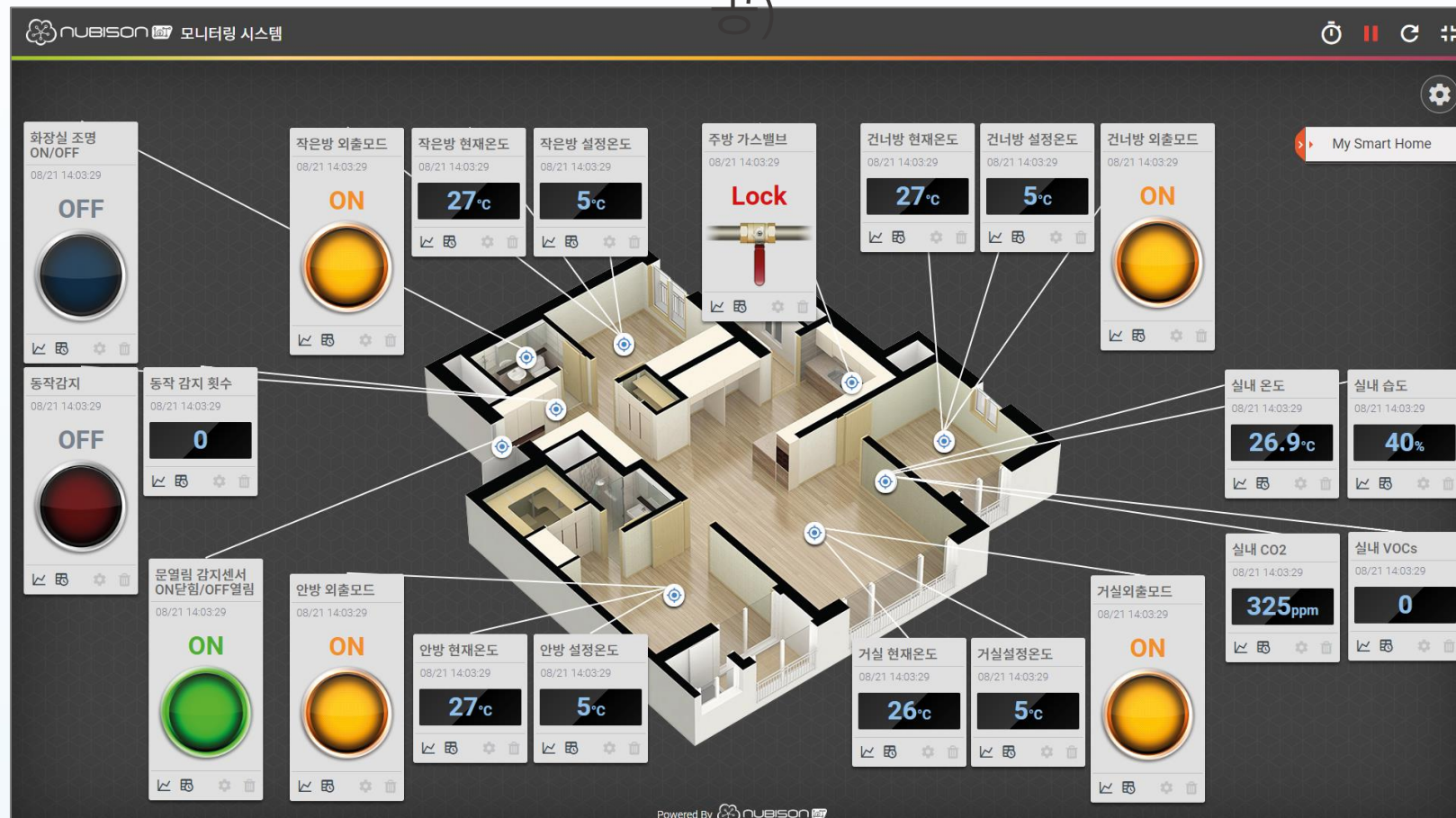


The screenshot displays the NUBISON IoT Rule Editor interface. The left sidebar shows a 'Rule 보드' (Rule Board) with various components like '물(GL\_R)', '논리', '컨트롤러(GL\_R)', '연산', '변수', and '알림(GL-R)'. The main workspace shows a rule configuration for temperature monitoring. The rule starts with '나만의 일 시작' (Start my own day) and includes several steps: '동작이름을 [냉장1\_유선온도센서21]로 명칭한다' (Set action name to [냉장1\_유선온도센서21]), '나의 냉장1(8°C ~ -5°C) 유선온도센서21의 가기를 00 명령으로 조회한다' (Check the value of my 냉장1(8°C ~ -5°C) 유선온도센서21 with the 00 command), '동작이름을 [냉장2\_유선온도센서22]로 명칭한다' (Set action name to [냉장2\_유선온도센서22]), and '나의 냉장2(8°C ~ -5°C) 유선온도센서22의 가기를 00 명령으로 조회한다' (Check the value of my 냉장2(8°C ~ -5°C) 유선온도센서22 with the 00 command). The rule then branches into two main paths: '동작실행 후 에러시엔' (After action execution, if error) and '정상실행시' (When running normally). The '정상실행시' path includes '연속작업을 시작한다' (Start continuous work), '작업 이름을 [냉장1\_유선온도센서21]로 명칭한다' (Set work name to [냉장1\_유선온도센서21]), and a series of conditional checks and actions based on the temperature sensor data. The '아니면' (Otherwise) path includes '물 이름 [냉장1\_유선온도센서21]의 알림 메세지 전송 횟수를 초기화 합니다' (Initialize the notification message transmission count of the 물 이름 [냉장1\_유선온도센서21]).



# 장소 모니터링

- 장소를 중심으로 한 디바이스 데이터의 시각화 화면 저작 및 실시간 모니터링 (차트/테이블 제공)





# 지도 기반 모니터링

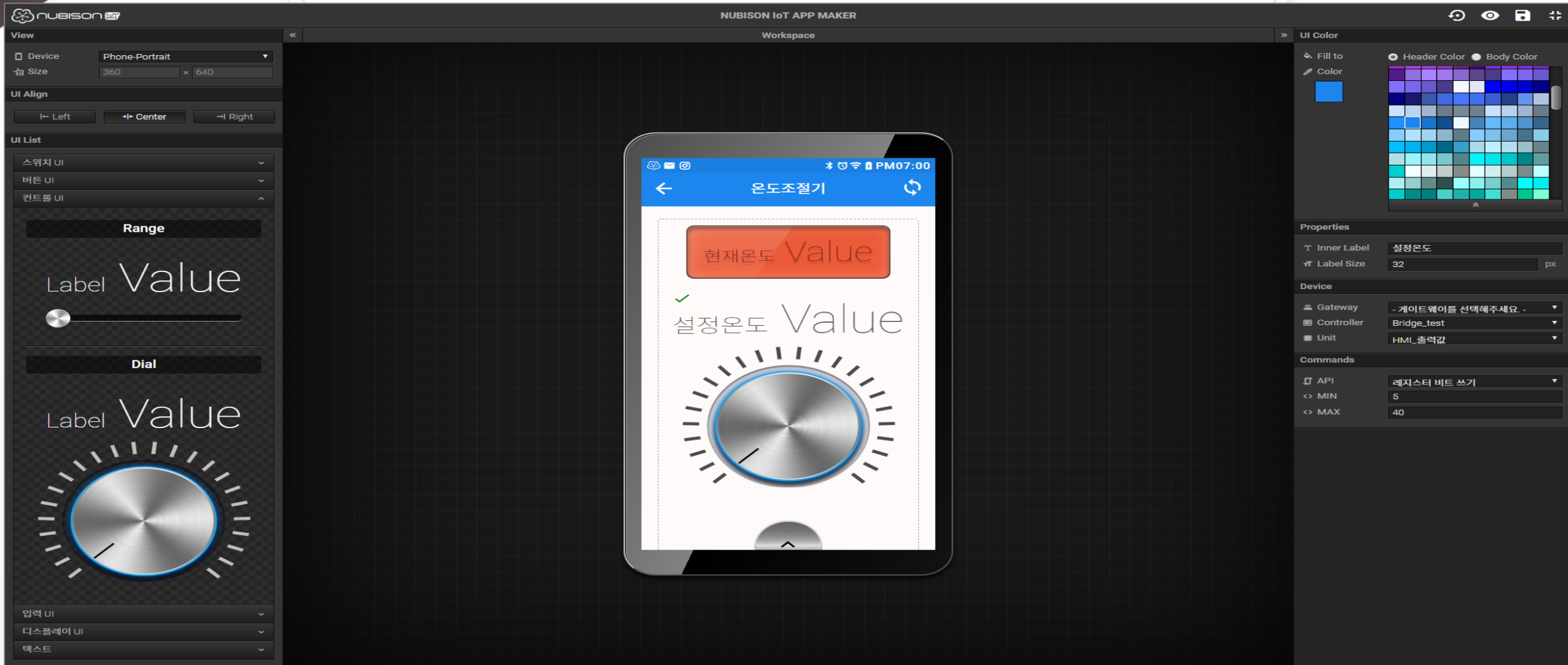
지도를 중심으로 한 디바이스 데이터의 시각화 화면 저작 및  
실시간 모니터링 (차트/테이블 제공)





# 모바일 앱 메이커

원격에서 디바이스의 데이터를 조회 및 제어가 가능한  
모바일 앱 화면 제작 툴






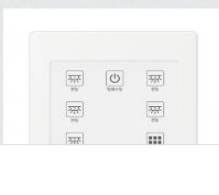








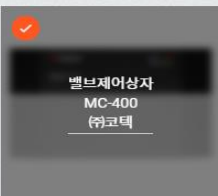







# 상용 컨트롤러

클라우드에 제품의 모델별 속성 및 구성정보를 등록하여 다중의 사용자가 쉽고 빠르게 사용할 수 있도록 가상의 클라우드 컨트롤러를 등록/배포

컨트롤러 추가

전체 브랜드

제품 검색



밸브제어상자  
MC-400 / (위코텍)

사용자 설정

컨트롤러 이름 \* 컨트롤러 이름을 입력하세요.

그룹 번호 \* 그룹 번호를 입력하세요.(0~16)

연결방식 선택 \* 연결방식을 선택하세요.

컨트롤러 정보

제품명 밸브제어상자

제품고유번호 MC-400

제조사 (위코텍)

드라이버 각방제어기

하드웨어 버전 MC-400

상세 설명

C-3100 밸브제어상자

주요사양

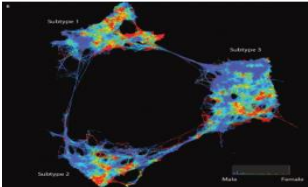

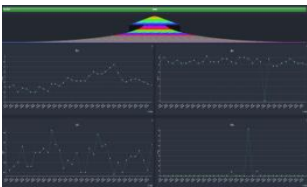
전원 : AC220V±10% 50/60Hz, 전원코드 3P-1M

온도조절기 : 최대 8개 연결

밸브구동기 : 8개까지(8개 추가 가능 - 확장보드 추가사양)

확인 닫기

실시간 IoT 데이터 분석 TDA(Topology Data Analysis), 주파수 분석, 머신러닝 분석 등 다양한 실시간 Data 분석 방법 제공

1 TDA	2 주파수 분석	3 머신러닝 분석
<ul style="list-style-type: none"><li>위상수학 기반 다변량의 데이터 형태화 모델링을 통한 분석 및 예측</li><li>분석결과 예시</li></ul>  <ul style="list-style-type: none"><li>- 데이터의 형상화 기술을 통해서 숨겨진 의미를 찾아냄</li></ul>	<ul style="list-style-type: none"><li>진동센서 데이터 분석을 통한 고장 예측 및 고장 위치 진단</li><li>분석결과 예시</li></ul>  <ul style="list-style-type: none"><li>- 3차원 입체적 Boundary를 설정, 기기 이상유무 판단</li></ul>	<ul style="list-style-type: none"><li>데이터 학습을 통한 분석 및 예측</li><li>분석결과 예시</li></ul>  <ul style="list-style-type: none"><li>- 데이터 구간 특징 학습을 통하여 실시간 예측</li></ul>

## 주요 특징 >

### ✓ TDA

- 다변량 데이터 분석을 통한 데이터 간의 연관 관계 분석
- Main Stream뿐 아니라, Minor Stream도 시각화 (Shape of Data)하여 분석하는 것이 가장 큰 특징

### ✓ 주파수 분석

- 3차원 입체 Boundary 이미지 제공
- 비정상(특정 부위 이상 발생) 주파수 분석에 대한 3차원 입체 Boundary 이미지 제공 및 머신러닝 학습 기능 제공

### ✓ 머신러닝

- 지도학습(회귀/분류), 비지도학습(클러스터링), 강화학습 으로 구성

# 누비슨 IoT의 활용



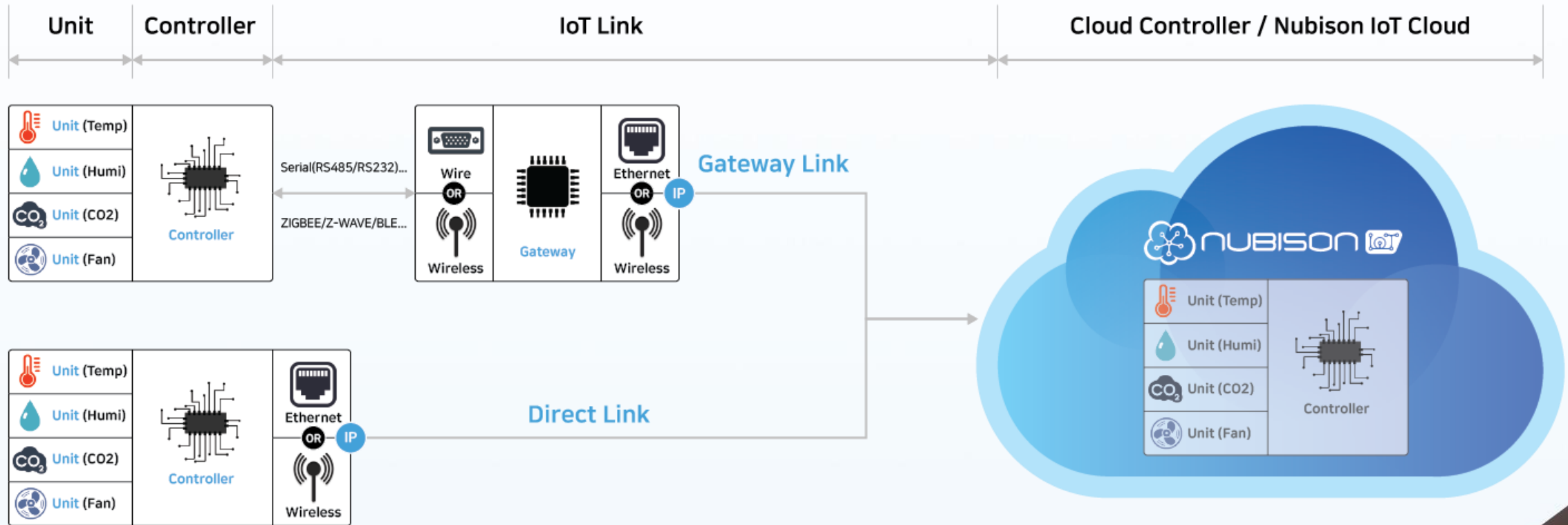
## 기본형 서비스



## 비즈니스 맞춤형 서비스

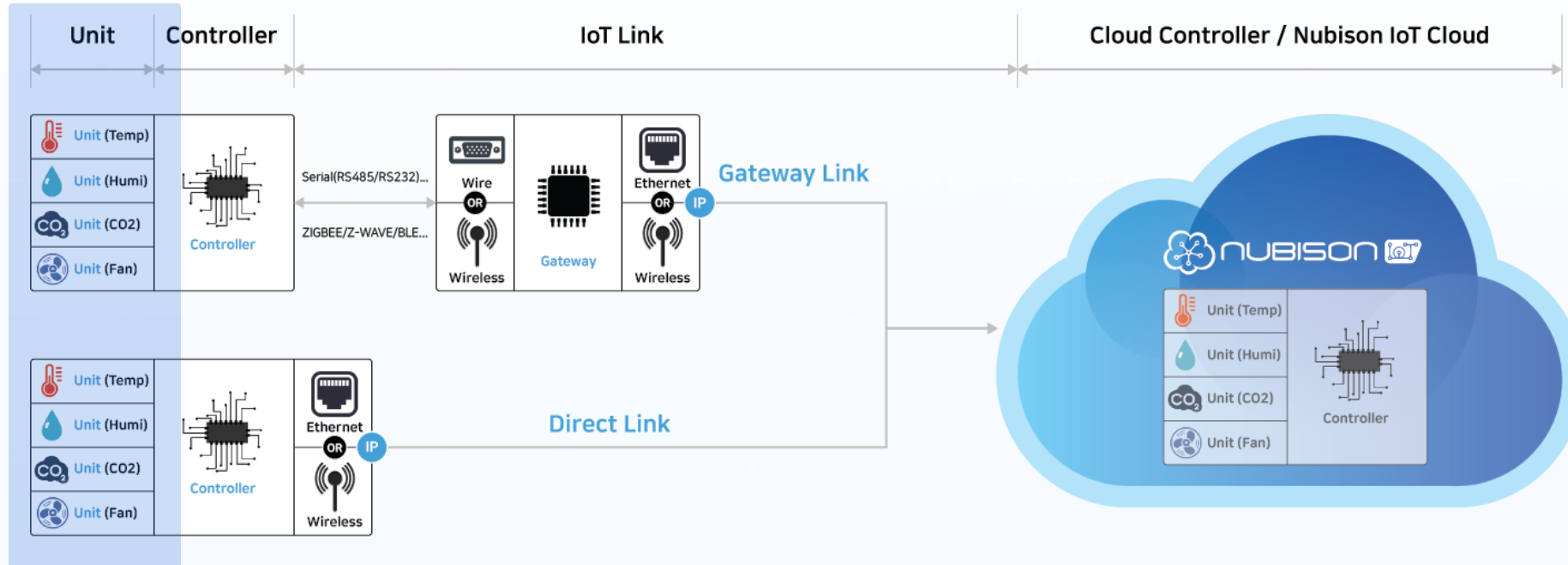


# 누비슨 IoT 구성



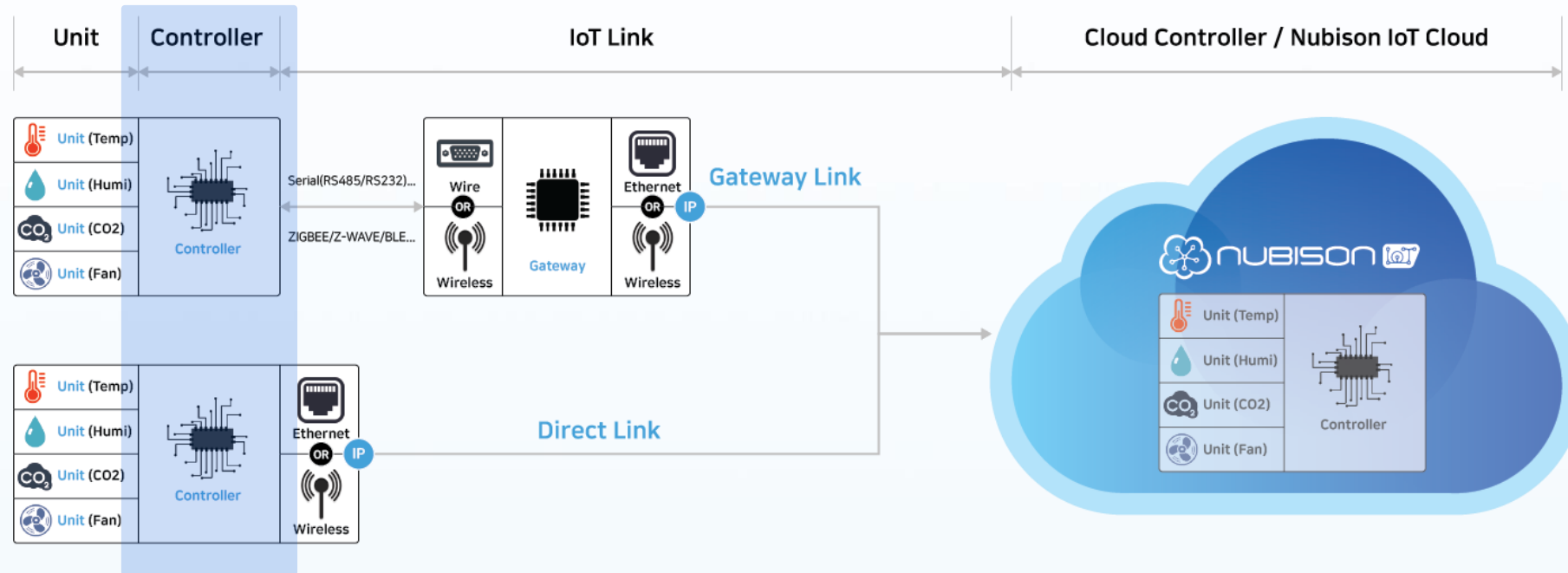


# 유닛 (Unit)



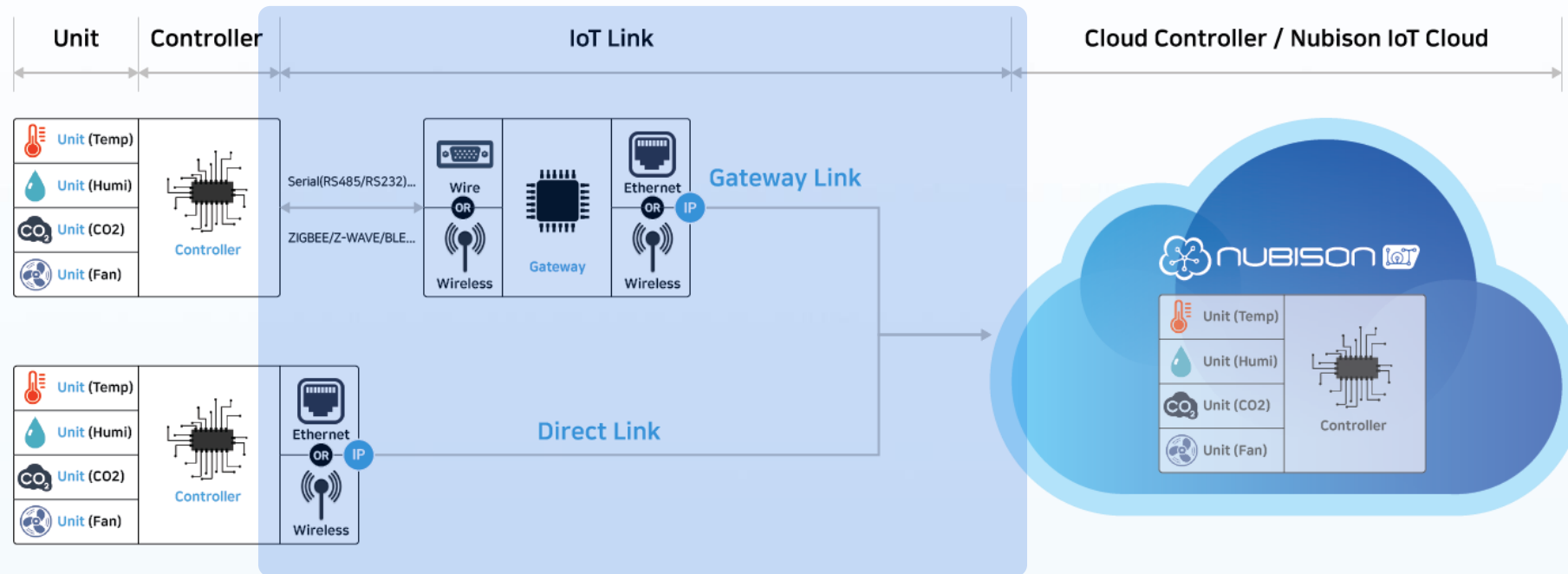
- 컨트롤러에 물리적으로 연결되어있는 센서 또는 동작기기를 지칭.
- 유닛은 **조회**만 가능하거나 **제어**만 가능할 수도 있고 혹은 **둘 다 가능**할 수도 있음.

# 컨트롤러(Controller)



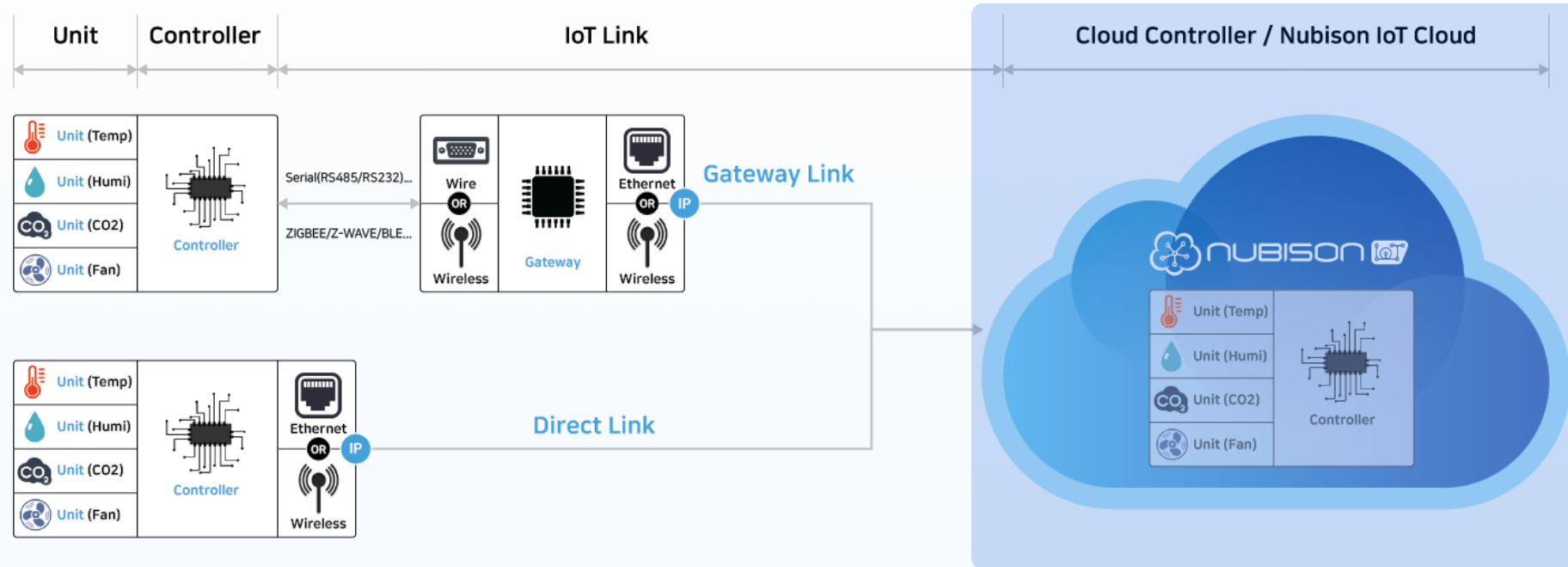
- 센서와 물리적으로 연결되어 있고 서버로부터 받은 제어 명령이나 데이터 조회 명령을 실질적으로 수행하는 하드웨어를 통칭.
- Gateway Link 하위에는 여러 컨트롤러가 존재할 수 있으며, Direct Link는 연결이면서 동시에 컨트롤러가 됨.

# IoT 연결 (Link)



- NUBISON IoT 클라우드와 연결되는 End Point를 지칭함.
- Non-IP 디바이스 같은 경우 게이트웨이를 통해 NUBISON IoT 클라우드에 연결될 수 있는데 이때는 게이트웨이가 연결이 되며 Gateway Link라 지칭함.
- IoT 디바이스 직접 NUBISON IoT 클라우드에 연결하는 경우 IoT 디바이스가 연결이 되며 Direct Link라 지칭함.

# 클라우드 컨트롤러



- 컨트롤러 템플릿으로써 클라우드 컨트롤러는 실제하는 컨트롤러의 가상화된 인스턴스임.
- 디바이스의 하드웨어적 특징이나 클라우드와 연결하는 프로토콜 및 드라이버 등의 설정을 가짐.



Explore NUBISON IoT

# Why Tizen OS?

## Why Tizen OS? (1/2)

MCU 성능 대비  
가격 하락 추세

기기 기능의  
복잡화

시스템 APP의  
검증 이슈

기기의 APP 및  
펌웨어 관리 이슈

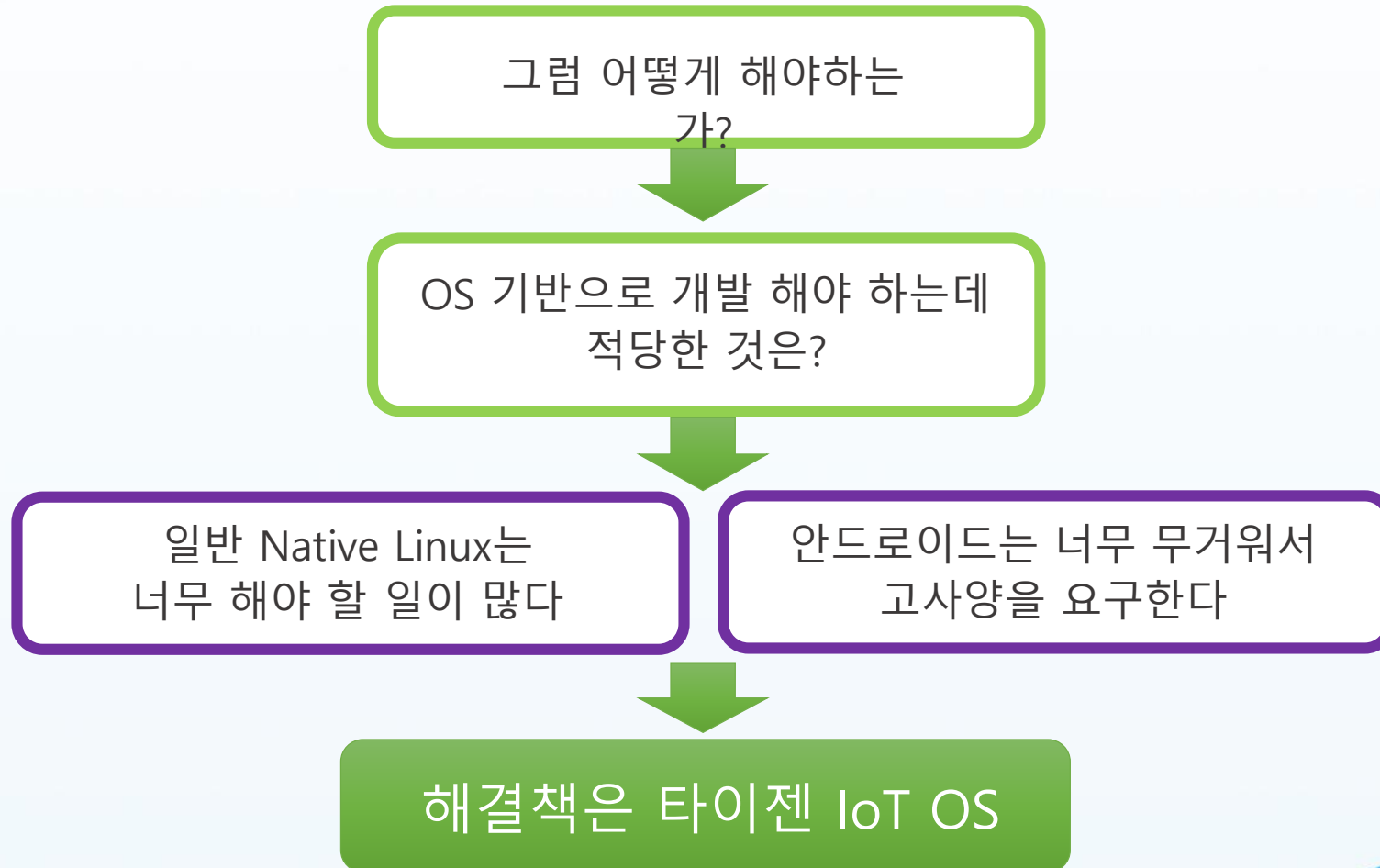
보안 및  
표준화 이슈



기존의 임베디드 방식 개발 및 RTOS로는 한계에 봉착!!



## Why Tizen OS? (2/2)

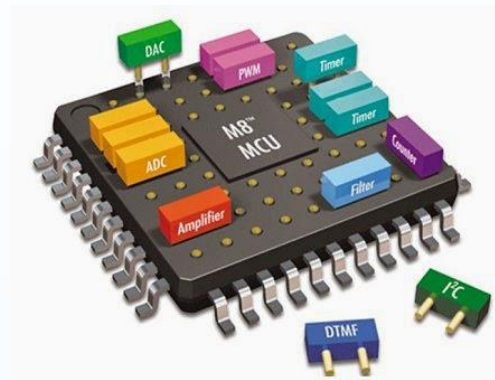


# 하드웨어 개발의 시작 (MCU이해)

기기들 마다 메인으로 컨트롤 하는 Chip이 있는데 그것을 보통 MCU라고 한다

(**마이크로컨트롤러**(Microcontroller) 또는 **MCU**(Micro Controller Unit)는 [마이크로프로세서](#)와 입출력 모듈을 하나의 칩으로 만들어 정해진 기능을 수행하는 컴퓨터를 말한다

위키에서)



그럼 MCU가 모든 일을 다하는것인가?



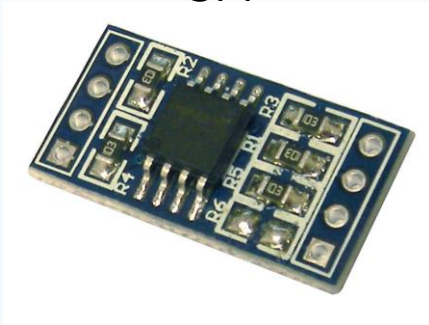
MCU는 말 그대로 메인 컨트롤을 담당하고 다양한 센서및 기기제어는 단위 모듈을 통해서 한다

# 하드웨어 개발의 시작 (인터페이스의 이해)

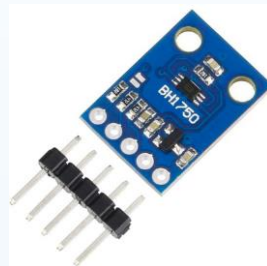
그럼 컨트롤 하는 방식은 어떻게 하는가?

MCU가 주변 모듈을 제어하는 방식은 크게 세 가지로 함

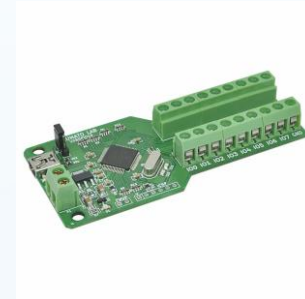
SPI



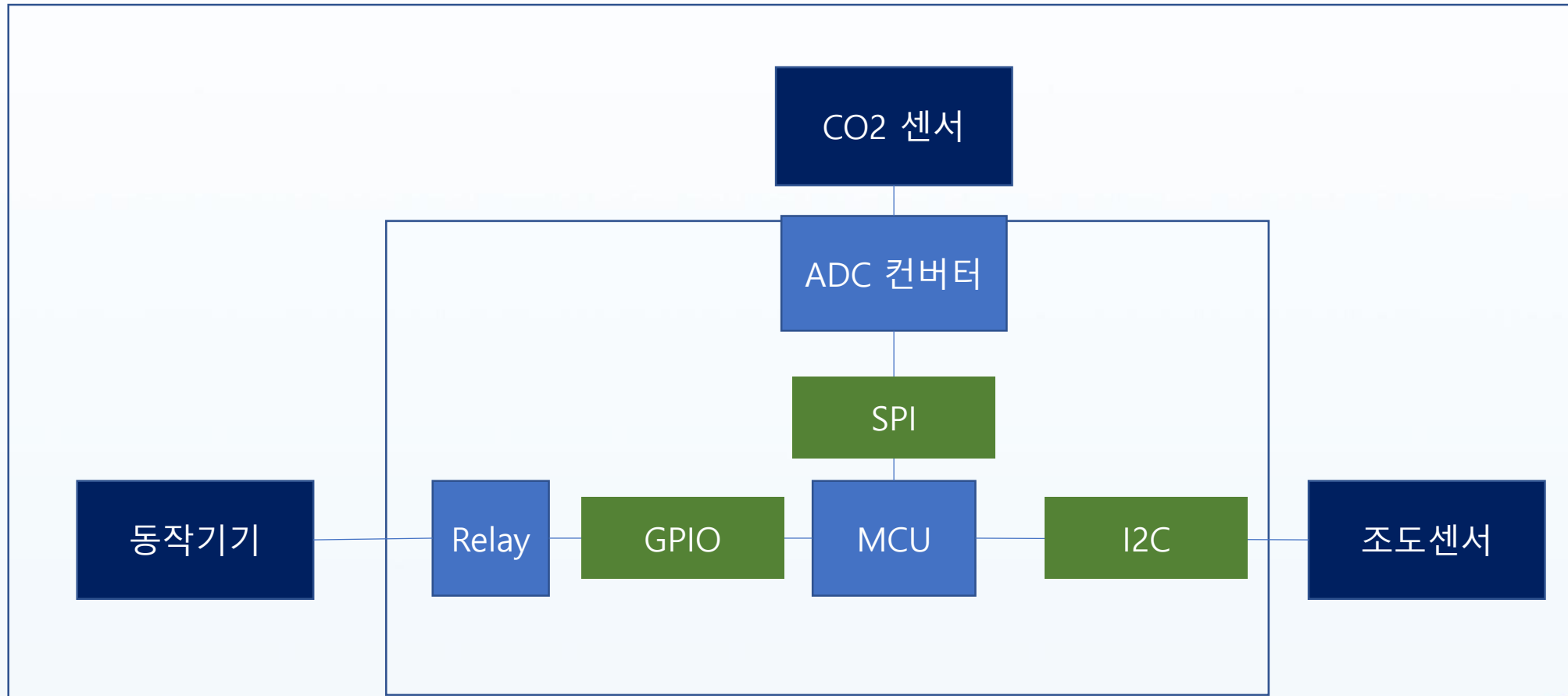
I2C



GPIO



# 하드웨어 개발의 시작 (실제 기기의 데모)



# 인터페이스의 상세 이해 (GPIO)

내용 추가 작성 예정

# 인터페이스의 상세 이해 (I2C)

내용 추가 작성 예정

# 인터페이스의 상세 이해 (SPI)

내용 추가 작성 예정



내용 추가 작성 예정